

Engineering Standard I-FRA-STD-001

Software Engineering

Document number: FRA-00050

This document is the property of Seqwater. It must not be copied or reproduced in any way whatsoever without the authority of Seqwater. This document is uncontrolled when printed. An electronic database manages and stores the controlled version.

Ver No.	Description	Document Owner		Document Approver		
		Position	Approval	Position	Approval	Approval Date
1	Amended Version	Team Leader Process & Analysis	Refer to supporting Info in REX	Manager Service Delivery	Refer to supporting Info in REX	11/06/2025

Contents

1.	Purpose	3
2.	Scope	3
3.	Groupings of Applicable Software.....	4
3.1.	Operational Technology (OT) Software	4
3.2.	Infrastructure as Code	7
3.3.	Other Developments.....	8
4.	Software Development.....	9
4.1.	Requirements Management.....	10
4.1.1.	Operational Technology	10
4.1.2.	Infrastructure as Code	11
4.1.3.	Other Developments	12
4.2.	Design.....	12
4.2.1.	Operational Technology	12
4.2.2.	Infrastructure as Code	13
4.2.3.	Other Developments	13
4.3.	Development & Verification	13
4.3.1.	Operational Technology	14
4.3.2.	Infrastructure as Code	14
4.3.3.	Other Developments	15
4.4.	Release & Deployment.....	15
4.4.1.	Operational Technology	15
4.4.2.	Infrastructure as Code	16
4.4.3.	Other Developments	17
4.5.	Operation & Maintenance	17
4.5.1.	Operational Technology	17
4.5.2.	Infrastructure as Code	18
4.5.3.	Other Developments	18
5.	Definitions	19
6.	References and Related Materials	20

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	2 of 21

This document is the property of Seqwater. It must not be copied or reproduced in any way whatsoever without the authority of Seqwater. This document is uncontrolled when printed. An electronic database manages and stores the controlled version.

1. Purpose

The purpose of this Framework is to define software engineering within Seqwater. This Framework is informed, aligned and developed from *ISO 12207 Systems and software engineering – Software life cycle processes and Guide to the Software Engineering Body of Knowledge (SWEBOK), V4.0*.

The definition of software engineering as fit for purpose for Seqwater is as follows:

Practices related to the development, operation and maintenance of software.

where software is defined as computer programs, procedures, documentation and data related to the operation of a computer system.

The requirement for development of software limits the application of this Framework to that where programming or coding is required. That is, this Framework will not apply to Commercial off the shelf (COTS) software products and / or software where configuration only is required.

2. Scope

This Framework applies to all workers and / or employees that are involved in software engineering related to the design, development and associated operation and maintenance of the software. The software relating to this Framework are those that are developed by or on behalf of Seqwater.

Software covered in this document includes:

- Any software used for internal use, whether produced by Seqwater or by a third-party (e.g. Programmable Logic Controller (PLC) code)
- Infrastructure as Code (IaC) deployed via Azure Pipelines.
- Any software developed that performs CRUD (Create, Read, Update, Delete) operations, especially CUD operations on data or files primarily using code. Integrations especially will trigger this requirement.

Software that isn't included within document is:

- Apps made within services (e.g., Power apps, forms, etc.) which is covered in [POL-00160](#) Citizen Development Management Policy.
- Applications created by an external organization (Microsoft applications, Siemens PLC programming applications, etc.).
- Any scripts or code blocks that are used to read or process data to present or visualise insights in embedded or separate dashboards or reports. Such code/script does not modify data.

This Framework outlines requirements for software development. Seqwater traditionally uses the Waterfall model that requires each step to be completed prior to the next step commencing. The Waterfall model is typically utilized when software cannot be easily changed and / or extensive testing is required. However, the requirements in this Framework, can be applied using other methodologies, such as Agile.

This Framework is aligned with the *ISO 12207-Systems and software engineering guidelines*. As per ISO 12207 section 4 conformance, selected processes are tailored to be fit for purpose for Seqwater software development, operation and maintenance. This therefore deems the conformance with ISO12207 as 'tailored conformance'.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	3 of 21

3. Groupings of Applicable Software

3.1. Operational Technology (OT) Software

Key Operational Technology (OT) software includes the documentation and application artefacts pertaining to Programmable Logic Controller (PLC), Supervisory Control and Data Acquisition (SCADA) systems and Historian systems. There is expected to be further OT software in the future. The structure of OT software is shown in Figure 1 and summarised as follows:

1. OT Standards: the following key standards are highlighted:
 - a. I-LST-STD-001 Control Systems Prescribed Equipment ([SPE-00342](#))
 - b. I-SPE-STD-005 High Priority Alarms & Interlocks Management ([SPE-00356](#))
 - c. I-SPE-STD-006 Alarm and Event Management ([SPE-00357](#))
 - d. I-SPE-STD-010 Control Systems Tag Naming ([SPE-00359](#))
 - e. I-SPE-STD-011 Control Systems Testing ([SPE-00360](#))
 - f. I-SPE-STD-013 Control Systems Design and Construction ([SPE-00361](#))
2. Functional Descriptions and functional specifications, which describe the function, features and limitation of equipment as configured using Seqwater library and how this is achieved.
3. PLC library
 - a. PLC block datasheets which describe the functionality of PLC blocks
 - b. Multiple PLC library blocks of code
 - c. Standard PLC Code forms the foundation of an operational PLC.
4. SCADA library
 - a. SCADA object datasheets which describe functionality of SCADA objects
 - b. SCADA templates including page graphics, objects and code to present information to the operator
 - c. SCADA data processing, transformation, report generation, and as needed trending tools.
 - d. SCADA library code, supporting advanced or customised functionality of the SCADA from out of the box functions, functionality, scripts and reporting.
5. Equipment Data Dictionary/Model
 - a. Definition of data available from each equipment
 - b. Selected data available from the field.
 - c. Definition of needed calculations from the data
 - d. Definition of standard acceptance, rejection and cleaning of data.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	4 of 21

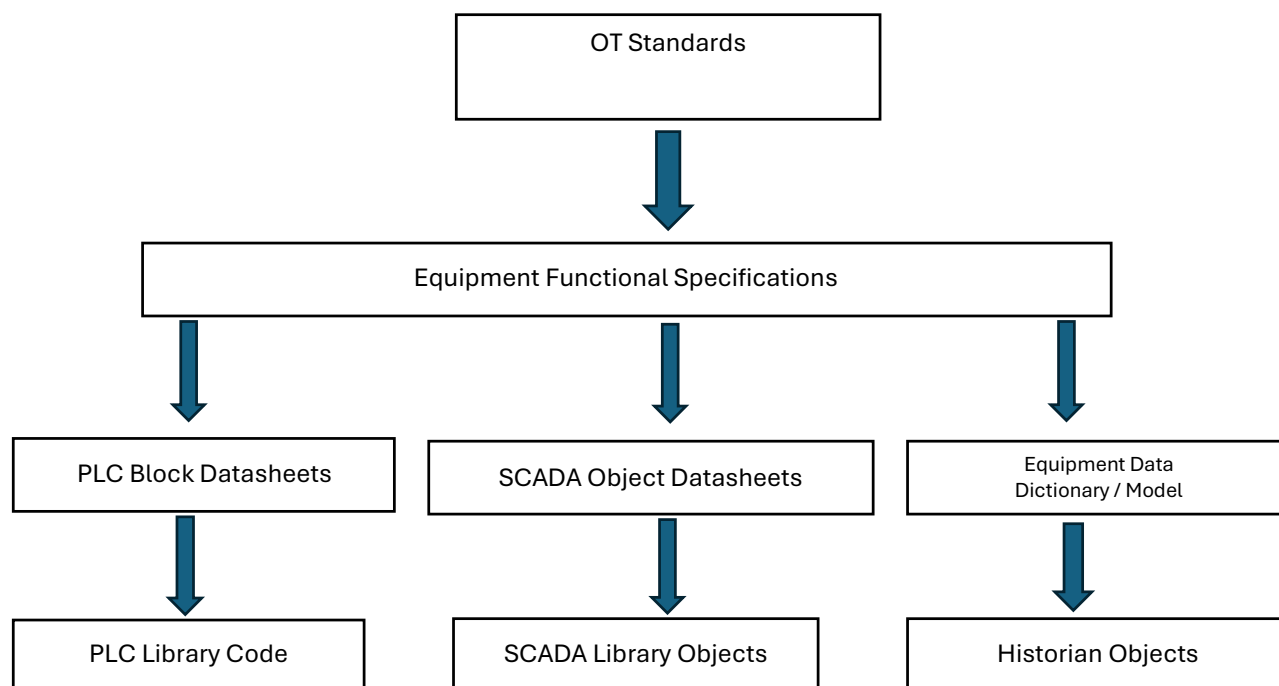


Figure 1 OT Software Structure

OT Standards provide requirements to achieve compliance with Acts, Regulations, Industry guidelines and standards and manage technical risk towards as low as reasonably practical. These inform the creation of technical documentation including specifications and datasheets.

The Equipment Functional Specifications describe the control of a device or group of devices. Equipment functional specifications shall prescribe a detailed template for each equipment. It shall describe how various library blocks/objects shall interact for a particular equipment. It shall also include the instrument calibration details such as upper and lower range limits, alarms, discrepancy actions and device standard I/O's and SCADA interface requirements.

PLC block/SCADA object datasheets shall pertain to a single block of code or object and provides a detailed description. Typically, many PLC blocks/objects are used for a single device and the assembly of these is defined in the equipment functional specification.

All functional specifications and datasheets are stored as per the procedure outlined in X-PRO-STD-008 Asset Standards Management and Application. That is, these are available as controlled documents on Waternet. An index of the relevant controlled documents are at the following location:

- On the Waternet home page select Asset Management then,
 - Seqwater Engineering Standards & Framework then,
 - Section: 5 – Operational Technology

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	5 of 21

This document is the property of Seqwater. It must not be copied or reproduced in any way whatsoever without the authority of Seqwater. This document is uncontrolled when printed. An electronic database manages and stores the controlled version.

OT Software branching and merging strategy

The adopted branching and merging strategy applied for OT software is shown diagrammatically in Figure 2.

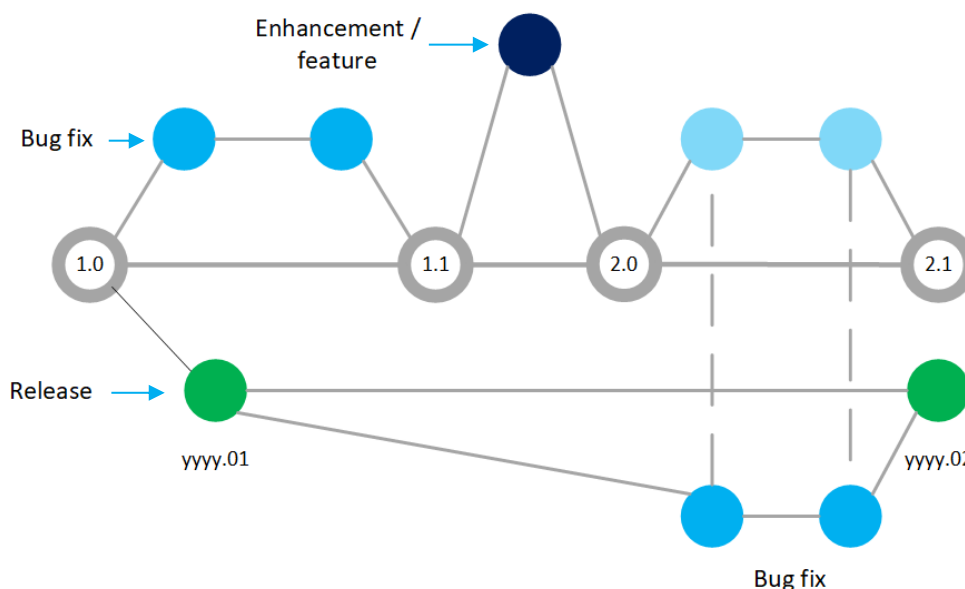


Figure 2 OT software branching and merging strategy

The following rules, with reference to Figure 2, are applied:

- A mainline of code acts as the official version including the latest and preceding versions (i.e. 1.0, 1.1, 2.0, 2.1). Versioning of the code are annotated with a major and minor release as a whole and partial number, respectively.
- There are defined 'releases' from the mainline (i.e. 1.0). These are numbered with a release value using the year and an incremental numerical value (i.e. yyyy.01).
- Corrective modifications (bug fixes) may be applied to a release and must be documented for evaluation for integration or rejection to the mainline. Branches in operation must be superseded by the mainline once a new release with the evaluated corrective modification is issued.
- Features and / or enhancements are only integrated to the mainline.
- Corrective modifications (bug fixes) may be applied to the mainline but would only appear in a new release.

The mainline library versions are recorded within the PLC block/SCADA object datasheets. Refer to I-REG-STD-002 Register for list of currently available standard PLC blocks and SCADA objects.

OT Software enabling systems

Enabling systems shall include:

- Requirements management software (Jira)
- Integrated Development Environments (IDE) products (Siemens Totally Integrated Automation (TIA) and Schneider Control Expert, PlantSCADA, AVEVA PI and PI Vision)
- Test systems and software (Test / training domain (SCADATest.local) Jira + Zephyr and Siemens SIMIT)

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	6 of 21

- Software Git repository and release management (Azure DevOps Repos¹)
- Deployment management (Test systems and potentially Octoplist)

Operations and maintenance of software is via existing systems as per [PLN-00388](#) Monitoring and Control Systems Asset Class Plan.

3.2. Infrastructure as Code

Infrastructure as Code (IaC) provides cloud computing resources using code, ensuring consistency, scalability, and automation. Seqwater predominantly uses Azure cloud hence this Framework is skewed towards implementing IaC using Bicep within Azure DevOps pipelines. Consistent with Seqwater's multi-cloud strategy, future versions of this Framework will alter / expand to include Terraform as a cloud-agnostic IaC implementation tool to achieve a maintained code base across Azure and AWS.

IaC branching and merging strategy

The Azure DevOps branching and merging strategy has been adopted for IaC is shown diagrammatically in Figure 2.

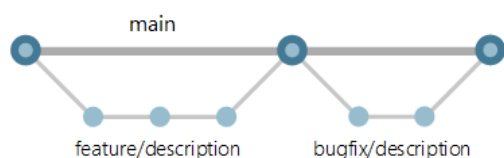


Figure 2 IaC branching and merging strategy ²

The following rules, with reference to Figure 2, are applied:

- The main branch of code acts as the official version for deployment to production.
- Separate from the main branch, create a feature or bugfix to address new functionality or defects.
- Prior to production deployment, merge the features or bugfixes into the main branch.

IaC enabling systems

Enabling systems shall include:

- Requirements management software (Jira)
- Test systems and software (development and user acceptance testing resource groups in Azure, Jira + Connectors + Zephyr and equivalent third party platform test environments involved in integrations as appropriate (MainPac, Ventia SAP as examples))
- Software Git repository and release management (Azure DevOps)
- Deployment management (Azure DevOps)

¹ Siemens AX is a Git product that will be assessed in the event Azure DevOps is deemed unsuitable. The risk with Azure DevOps is that it does not integrate natively with the IDE products. This risk will be appraised with implementation of this framework.

² From <https://learn.microsoft.com/en-us/azure/devops/repos/git/git-branching-guidance?view=azure-devops>
 Article dated 3/06/2024, accessed 14/04/2025

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	7 of 21

IaC will be delivered via Azure Pipelines in the following four-stage workflow:

6. Build Stage

- Validate and compile Bicep files into Azure Resource Manager (ARM) templates.
- Perform syntax and linting checks to ensure correctness.
- Store artifacts for subsequent deployment stages.

7. Dev Stage

- Deploy infrastructure to a development environment.
- Run initial validation tests.
- Allow developers to test application compatibility.

8. Test Stage

- Deploy to a staging environment that mimics production.
- Execute automated infrastructure and security tests.
- Perform performance and integration testing.

9. Prod Stage

- Deploy to the production environment.
- Ensure approval via Azure DevOps Environments.
- Implement rollback mechanisms in case of failure.

The requirements as specified in this Framework is overlaid with the Azure Pipelines processes as described above.

3.3. Other Developments

Other developments are characterised by the CRUD functions as per Section 2. Examples may include use of Python, Fortran, Pascal or Basic utilized to build or customize applications.

All of these should adhere to the following guidelines and requirements.

- Documents or specification to support development of the desired end result.
- Distinct development, testing and production environments.
- Versioning and retention of code within an Azure DevOps project provisioned by DTI Service Delivery.
- Documented test cases and results.

Development in other languages should also ensure that appropriate governance as outlaid in this document is adhered to.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	8 of 21

4. Software Development

Table 1 lists the 14 technical processes noted in ISO12207 and consolidates and aligns these to groupings of applicable software to outline how ISO12207 is applied in Seqwater.

Table 1 ISO12207 technical processes mapped to groupings of applicable software

ISO12207 Technical Processes	Infrastructure as Code	Operational Technology	Other Developments
Business or Mission Analysis	Requirements Management	Requirements Management	Requirements Management
Stakeholder Needs and Requirements Definition			
System/Software Requirements Definition			
Architecture Definition	NA	NA	NA
Design Definition	Design	Design	Design
Systems Analysis	NA	NA	NA
Implementation	Development & Verification	Development & Verification	Development & Verification
Integration			
Verification			
Transition	Release & Deployment	Release & Deployment	Release & Deployment
Validation			
Operation	Operation & Maintenance	Operation & Maintenance	Operation & Maintenance
Maintenance			
Disposal	Disposal*	Disposal	Disposal

Application of these processes is required regardless of the implementation methodology, waterfall or an agile method, adopted. There may be differences in the extent to which the process is deemed complete before starting another process and the extent of iterations.

*In relation to IaC, Disposal is both Seqwater's preferred approach of disposing of development and user acceptance environments at the end of a project. If environments are required, they can be redeployed for a future project phase. If production systems in Azure are no longer required, they may also be disposed of.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	9 of 21

This document is the property of Seqwater. It must not be copied or reproduced in any way whatsoever without the authority of Seqwater. This document is uncontrolled when printed. An electronic database manages and stores the controlled version.

The key activities and outputs from each of the processes are outlined in the following sections.

4.1. Requirements Management

All software shall be aligned to a, or multiple, Seqwater strategy(ies) or business objective(s).

Software will be constrained to nominated systems. This may include exclusion of legacy systems or non-compliant systems that are inconsistent with a defined policy or standard. Software may be constrained by critical quality or compliance drives, these may include but are not limited to cyber security, process safety, etc.

Stakeholders' identification shall include mapping of persons or teams for impact and influence. This shall inform a stakeholder engagement and communications plan. Stakeholder management shall be required for the life of the software.

At a minimum, stakeholders representing the following groupings shall be represented:

- Users: defined as those that will use the software to perform their tasks.
- Customer: defined as those that define requirements for the software and typically are users as well.
- Sponsor: defined as those who authorize the software.

Stakeholders needs must be captured, analysed and written as specific measureable requirements and validated. Requirements must be prioritized and planned for implementation. Any requirement contrary to Seqwater strategy or business objectives will be deemed void. Stakeholders must provide endorsement of the requirements and the sponsor will need to approve. Where requirements are conflicting across stakeholders, there shall be a method to resolve. Requirements can exist for the life of the software and must be traced through to the completed product. All requirements shall have a specified review date, nominated at the requirements approval, where these are reviewed, re-endorsed and re-approved. Requirements should be managed using change control processes. Seqwater prefers the use of Jira for requirements management.

Critical performance measures shall be defined and will be used to assess the product as completed. These may be updated at the start of each release cycle.

4.1.1. Operational Technology

Operational Technology software shall align with Seqwater's Operational Transformation strategy (contained in *Seqwater Corporate Strategy 2024 – 2028*) and Seqwater's OT architecture strategies and guidelines ([POL-00116](#) Enterprise Architecture Policy including referenced documents within).

Traceability of strategic business objectives shall be maintained for the life of the software.

A key constraint is that Seqwater will not invest in developing software for legacy or non-prescribed products (as defined in I-LST-STD-001 Control Systems Prescribed Equipment ([SPE-00342](#))). These shall seek a dispensation from Seqwater cyber security standards due to increased risk by not attaining the required versioning level of service (i.e. N-2). These systems will need to be upgraded to facilitate operation with Seqwater's OT software.

Requirements can be elicited from:

- Defects and enhancements captured in lodged (I-REG-STD-002 PLC Std Block Data sheet and Test Reports Document Register and I-REG-STD-004 SCADA Std Objects Data Sheet and Test Reports Register)
- Legislation or Standards revisions, including all international, national and Seqwater own standard documents. Safety, security, environment and water quality requirements must be specified and may be informed by legislation or international standards.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	10 of 21

- Specific workshops with identified stakeholders
- Other inputs from external parties to Seqwater such as contractors, peer organisations, IDE software vendors, etc.

Demonstrations and example build will be used to help convey technical requirements to user experiences. The test environment, SCADAtest.local, will be used for this purpose. These example builds are not intended to be designs but instead a communications tool. Use cases or user stories may also be used as a technique to help capture requirements.

Key users and customers of Seqwater's OT software are as follows:

- Control System Engineers
- Process Engineers, energy specialists and water quality scientists
- Operations Supervisors / Co-ordinators
- Cyber security

The sponsor should be from the operations management line.

Requirements will be analysed, framed and captured in business speak with defined measurement using Jira. Requirements should be at least classified as functional and non-functional. Any requirement that does not align with Seqwater strategic business objectives will be deemed void. Where requirements are conflicting amongst stakeholders, these shall be assessed using a risk-based assessment. Requirements will be prioritized (mandatory versus preferred or desirable) and allocated to a software release cycle.

Stakeholders will be requested for endorsement of requirements, what is defined as in scope or out of scope for each particular release, and the release schedule. The sponsor shall ultimately provide approval and have the authority to override any or all stakeholders.

The requirements' gathering process is continuous but the requirements framing and scheduling is due twice per year, aligning with the software release cycle.

Requirements shall be traced through to specific test cases, as further described throughout this Framework.

Critical performance measures particularly aligned with mandatory requirements shall be defined at the start of each release cycle. These shall be Specific, Measurable, Attainable, Relevant and Timely (SMART) specifications. The software product from implementation of the development cycle shall be assessed against these performance measures to be deemed complete.

4.1.2. Infrastructure as Code

IaC deployments shall align with Seqwater's architecture strategies and guidelines ([POL-00116](#) Enterprise Architecture Policy including referenced documents within).

Traceability of strategic business objectives shall be maintained for the life of the software.

A key constraint is that Seqwater will not invest in deployments for legacy or non-prescribed products. These shall seek a dispensation from Seqwater cyber security standards due to increased risk by not attaining the required versioning level of service (i.e. N-2). If approved these products may be deployed in a containerised fashion.

Requirements shall be collated from:

- Specific workshops with identified stakeholders

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	11 of 21

- Other inputs from external parties to Seqwater such as contractors, peer organisations, software vendors, etc.

Potentially all users of Seqwater IT systems and data derived from integration services are stakeholders of solutions deployed via IaC.

The sponsor shall be DTI Service Delivery manager.

Requirements will be analysed, framed and captured in business speak with defined measurement using Jira. Requirements should be at least classified as functional and non-functional. Any requirement that does not align with Seqwater strategic business objectives will be deemed void. Where requirements are conflicting amongst stakeholders, these shall be assessed using a risk-based assessment. Requirements will be prioritized (mandatory versus preferred or desirable) and allocated to a software release cycle.

Stakeholders will be requested for endorsement of requirements and the release schedule. The sponsor shall ultimately provide approval.

The requirements' gathering process is continuous but will largely be derived from projects or operations as required.

Requirements shall be traced through to specific test cases, as further described throughout this Framework.

4.1.3. Other Developments

Deployments in other languages shall align with Seqwater's architecture strategies and guidelines ([POL-00116](#) Enterprise Architecture Policy including referenced documents within).

Traceability of strategic business objectives shall be maintained for the life of the software.

A key constraint is that development and product will comply with Seqwater cyber security process and standards:

- [POL-00091](#) Information Security Management Policy
- [SPE-00531](#) Cyber Security Assurance Specification

Requirements must be gathered and documented in a formal repository adhering to section 4.1.

4.2. Design

The design will transform the requirements into design outputs which may include but is not limited to:

- Design characteristics or design basis or similar document / specifications e.g. security considerations, availability, fault tolerance, scalability, etc.
- Specification of systems e.g. IDE software, programming language, standards, repositories, etc.
- Design outputs e.g. database structures, technical specification documents, integrations, user roles, etc.

For engineering design, this must be undertaken by a Registered Professional Engineer Queensland (RPEQ) within the appropriate engineering area of practice.

4.2.1. Operational Technology

Key design output artefacts are shown in Section 3, Figure 1 as follows:

- Functional specifications: documentation and supporting example builds to describe an object design
- Datasheets & blocks and objects: documentation that specifies the code as well as the actual code / object

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	12 of 21

This document is the property of Seqwater. It must not be copied or reproduced in any way whatsoever without the authority of Seqwater. This document is uncontrolled when printed. An electronic database manages and stores the controlled version.

Functional specifications and datasheets are denoted as engineering documentation and must be created by and signed by an appropriately skilled RPEQ engineer. Management of these engineering documentation will adhere to Seqwater engineering management processes (X-PRO-STD-008 Asset Standards Management and Application Procedure ([PRO-02205](#))).

The design must meet technical specifications translated from requirements. Requirements that are mandatory must be implemented in the release cycle. Requirements that are preferred or desired may be deferred to later release cycles.

In addition, the design will be informed by requirements contained within standards and specifications, including but not limited to:

- Siemens / Schneider / AVEVA programming guides and other guidelines and documents.
- Alarming design as defined in I-SPE-STD-006 Alarm and Event Management ([SPE-00357](#))
- Functional requirements as defined in I-SPE-STD-005 High Priority Alarms & Interlocks Management ([SPE-00356](#))
- Coding standards is further defined in I-SPE-STD-013 Control Systems Design and Construction ([SPE-00361](#))
- Checks, quality control and test documentation as per I-SPE-STD-011 Control Systems Testing ([SPE-00360](#))

4.2.2. Infrastructure as Code

Key design output artefacts as follows:

- README and Change Log stored against Azure DevOps Repo.
- Jira cards, manually created.
- Jira cards, automatically created via the Jira Azure DevOps Connector linked to all Azure DevOps deployment events.

The design will be informed by requirements translated into technical specifications. Requirements that are mandatory must be implemented in the release cycle. Requirements that are preferred or desired may be bumped to later release cycles.

4.2.3. Other Developments

Key design output artefacts should include:

- Formal specification to support the code artifact, this should have quality checks demonstrating record versioning and be readily accessible.
- Specification of systems with which the code interacts, as an example Microsoft SQL database and version thereof.
- Documented design outputs such as database structures and user roles.

4.3. Development & Verification

Development shall be done consistent with the processes and procedures as defined in the Design phase. A software product, as informed from the Design is developed as the output.

There shall be appropriate change management, repository and review processes of software items. A modern software development platform, Git, will provide best practice systems.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	13 of 21

Verification will demonstrate requirements have been achieved with the software product. Jira is the preferred product used for requirements management and integrates with verification testing via Zephyr. The Design should have included key test cases and these may be expanded and added to for completeness and then executed.

4.3.1. Operational Technology

Azure DevOps Repos is used in OT software development. The processes for change management, repository and reviews contained in Azure DevOps are best practice and applied as recommended. Records from Azure DevOps are to be retained as development records.

IDE software from suppliers as defined in I-LST-STD-001 Control Systems Prescribed Equipment ([SPE-00342](#)) are used for development. All development of OT software is compliant with the standards, procedures, processes and policies as outlined in the Design.

SCADATest.local or equivalent offline environment shall be used as development environment.

The output shall be standardised objects used within the OT technologies i.e. PLC, SCADA or historian.

Using Jira and Zephyr, applicable requirements shall be written as test cases and executed. Demonstration tests, with stakeholders, will be the main method for testing using the SCADATest.local environment. Test cases must achieve a PASS and may require iteration with development to achieve this. Where a test case cannot be performed, these shall be noted and risk assessed. Where a test case is a FAIL and cannot be altered from further development activities, these shall be noted and risk assessed (e.g. absence of physical equipment with associated inputs to execute a complete test scenario).

The verification test results, which can be extracted from Jira, will be used as records. The standards update and release processes as defined in X-PRO-STD-004 Development and Review of Asset Standards Procedure ([PRO-01874](#)) and X-PRO-STD-008 Asset Standards Management and Application will be used for software release.

4.3.2. Infrastructure as Code

Azure DevOps Repos are used in IaC development. The processes for change management, repository and reviews contained in Azure DevOps Repos are best practice and applied as recommended. Records from Azure DevOps Repos are to be retained as development records.

Distinct resource groups will be utilized for development and deploying in Azure, with clear segregation between development, test and production stages.

Using Jira and Zephyr, applicable requirements shall be written as test cases and executed. Testing shall be conducted by DTI, vendor or end users as appropriate in the test Azure resource group. Test cases must achieve a PASS and may require iteration with development to achieve this. Where a test case cannot be performed, these shall be noted and risk assessed. Where a test case is a FAIL and cannot be altered from further development activities, these shall be noted and risk assessed (e.g. end to end test environment not available, an example would be our Entra/M365 instance).

The verification test results, which can be extracted from Jira, will be used as part of Release processes of the software product.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	14 of 21

4.3.3. Other Developments

Azure DevOps Repos are used in other language developments. The processes for change management, repository and reviews contained in Azure DevOps Repos are best practice and applied as recommended. Records from Azure DevOps Repos are to be retained as development records.

A non-production environment shall be requested from and deployed by DTI. This environment shall be utilised for any development.

Test cases which are traceable to the requirements with outcomes will be documented and stored in a formal repository. Test cases must achieve a PASS prior to release.

4.4. Release & Deployment

In compliance with the adopted branching and merging strategy for the group of software, a software version should be Released. A frequency of Releases shall be specified.

Test / User Acceptance Test (UAT) environments should be used. Documentation of test results must be retained. There should be traceability to software requirements.

Once Release is completed; deployment into an operating environment can progress. Deployment may be a single or many instances. Change management processes must be used. Key change management processes in Seqwater includes:

- [PRO-01816](#) ICT Change Management Procedure
- [PRO-02234](#) Management of Change (Technical) Procedure

Acceptance tests shall be required. Additionally, there may be proof of performance testing required. Any defects shall be captured on a punch list and be rated on impact. These defects must be managed and may contribute to new requirements as part of the software development lifecycle. The measure of complete shall be determined based on business objectives and requirements definition. Traceability to requirements is therefore critical.

4.4.1. Operational Technology

4.4.1.1. OT software release

OT software release shall correspond to two (2) release cycle per year, nominally in June and December. Requirements that are included in these releases will have been nominated from requirements management. Any requirements that could not be included in the release will be highlighted. A release will be numbered separately from the mainline code numbering.

The release shall be tested to demonstrate its suitable for the operating environment. The requirements as per I-SPE-STD-011 Control Systems Testing ([SPE-00360](#)) through to the end of Factory Acceptance Tests (FAT) should be completed. In addition, demonstration tests of selected operating assets, such as Water Treatment Plants (WTP) or bulk water pipelines, should be completed. These demonstration tests shall function to engage with end users particularly operators and the user interfaces. Preference is to use digital twins mimicking the selected assets as far as practicable to the operating system(s).

Release practices shall follow X-PRO-STD-004 Development and Review of Asset Standards Procedure ([PRO-01874](#)) and X-PRO-STD-008 Asset Standards Management and Application.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	15 of 21

4.4.1.2. OT software deployment

Deployment shall be managed via the OT change request process as per [PRO-02234](#) Management of Change (Technical) Procedure. Key considerations include:

- Management of downtime required for deployment of new software.
- Roll back procedures to mitigate risks of failed deployment.
- Test requirements shall be complete as per I-SPE-STD-011 Control Systems Testing ([SPE-00360](#)).
- Documentation completed including updates to all controlled documents as per I-SPE-STD-019 Process Control Systems Information Maintenance ([PRO-02329](#)), training and induction of operators and all other users.
- Performance as a measure of success should relate to requirements being achieved.

Acceptance as complete shall be via approval through the change management process.

4.4.2. Infrastructure as Code

4.4.2.1. IaC software release

IaC deployment shall occur as required during project phases and operations and as required to meet operational requirements.

The release shall be tested to demonstrate its suitable for production usage. Testing will be conducted with resources deployed to each staged resource group and connecting to equivalent end points in third party platforms. Preference is for testing to be conducted end to end but it is acknowledged this is not always possible, as an example Seqwater has a single M365/Entra tenant.

Release of software shall be first to a development resource group, then a test resource group for testing and on acceptance to a production resource group.

4.4.2.2. IaC software deployment

IaC software deployment shall be to the Seqwater South East Azure Region in appropriate resource groups. End to end testing will be completed when technically possible. Deployment shall be managed via the IT change request process delivering all mandatory deliverables as detailed in Change Management Information Sheet - Mandatory Deliverables for Normal Changes ([D24/206099](#)). Key considerations include:

- Management of downtime for production systems.
- Roll back procedures to mitigate risks of failed deployment. This may be automated as part of the IaC pipeline.
- Documentation completed as per Change Management Information Sheet - Mandatory Deliverables for Normal Changes ([D24/206099](#)).
- Successful testing of production deployment.

Acceptance as complete shall be determined by the sponsor from the DTI Service Delivery management line.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	16 of 21

4.4.3. Other Developments

4.4.3.1. Other Development software release

Other Developments deployment shall occur as a defined release instance.

The release shall be tested to demonstrate its suitability for production usage. Testing will be conducted in the test environment to ensure acceptance by users prior to production deployment.

4.4.3.2. Other Developments software deployment

Other Developments software deployment shall be to the appropriate production environments. Deployment shall be managed via the IT change request process delivering all mandatory deliverables as detailed in Change Management Information Sheet - Mandatory Deliverables for Normal Changes ([D24/206099](#)). Key considerations include:

- Management of downtime for production systems.
- Roll back procedures to mitigate risks of failed deployment.
- Documentation completed as per Change Management Information Sheet - Mandatory Deliverables for Normal Changes ([D24/206099](#)).
- Successful testing of production deployment.

4.5. Operation & Maintenance

Operation of software shall be by Seqwater business teams or units. The users must have been trained as part of the preceding deployment phase.

The software performance shall be measured and regularly reported against defined service levels.

Customer and user support shall be provided.

The software system shall be maintained as per standardized practices. Modifications, including defect rectification, will be systemized and use change control practices.

4.5.1. Operational Technology

Service levels shall be as specified in [PLN-00388](#) Monitoring and Control Systems Asset Class Plan. All maintenance will be performed in accordance with this plan, including the patching and update of software systems, such as IDE software or platform software. Software back-ups of deployed code will be retained in sub-versioning system (e.g. SVN).

Measurement will be near real time against these service level measurements. Measurement shall be provided on a dashboard available from within Seqwater systems.

Customer and user support shall be provided by:

- PLC and SCADA – Control Systems Maintenance (CSM) reactive maintenance
- Historian – DTI Service Delivery

CSM reactive maintenance will be by Mainpac reactive maintenance process (via Waternet, Operations and Maintenance). While DTI services will be by ITIL processes involving a service ticket raised (via Waternet, Business Support under Digital Technology & Information).

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	17 of 21

Where modifications to software are required, this shall be raised by an engineering [TEM-00215](#) Request for Engineering Services Template. This may include defect rectification. Simple bug fixes may be managed via these tickets but must comply with the branching and versioning strategy.

New features, enhancements and defect fixes will typically feed into new requirements and recorded in *I-REG-STD-002 PLC Std Block Data sheet and Test Reports Document Register* and *I-REG-STD-004 SCADA Std Objects Data Sheet and Test Reports Register*.

4.5.2. Infrastructure as Code

Service levels shall meet the DTI minimum SLA. All maintenance will be performed in accordance with this Framework. All code and changes thereof will be retained in Azure DevOps with clear commentary.

Measurement will be near real time against these service level measurements. Measurement shall be provided on a dashboard available from within Seqwater systems.

Customer and user support shall be provided by:

- DTI Cloud Services
- Outsourced vendors where such an agreement is in place as an escalation point.

While DTI services will be by ITIL processes involving a service ticket raised (via Waternet).

Where modifications to software are required, this shall be raised by a service ticket noting the applicable software product. This may include defect rectification. Simple bug fixes may be managed via these tickets but must comply with the branching strategy.

New features, enhancements and defect fixes will typically feed into new requirements and recorded in a Jira ticket.

4.5.3. Other Developments

Operation of software shall be as agreed per instance of development and system. This agreement must be prior to commencing development rather than informed on completion.

The software performance shall be measured and regularly reported against defined service levels by the software custodian.

Support of software shall be as agreed per instance of development and system. This agreement must be agreed prior to commencing development rather than informed on completion.

Maintenance of software shall be as agreed per instance of development and system. This agreement must be agreed prior to commencing development rather than informed on completion.

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	18 of 21

5. Definitions

Term	Definitions
Application	Refers to applications that are fully designed and implemented solutions. Seqwater has no control over the application or rights to modify it. Examples of this include. • Microsoft products • TIA portal • VM-Ware
Architecture	A blueprint for Seqwater's IT systems. Consists of framework, guidelines, templates and patterns to guide a solution to integrate with the business and existing technology design.
Client	A group or individual within Seqwater that requests for a software development project.
Computer System	A combination of interconnected hardware and software components that work together to process, store, and output information
Control systems	A system that responds to input signals from the equipment under control and/or from an operator and generates output signals that cause the equipment under control to operate in the desired manner. Components of control system include but are not limited to PLC, SCADA, network switches, fibre optic or Ethernet communication network, engineering station, operation station, software server, IO modules, remote IO nodes and control panels.
DevOps	A set of practices, tools, and a cultural philosophy that integrates software development (Dev) and IT operations (Ops) to streamline the process of delivering software.
Employees	Means persons employed directly by Seqwater in a permanent, temporary or casual capacity.
IDE	An integrated development environment (IDE) is a software application that provides comprehensive facilities for software development.
Operational Technology (OT) Systems	Technology that interfaces with the physical world and includes Control Systems as a subset, which in turn includes Supervisory Control and Data Acquisition (SCADA) and Programmable Logic Controllers (PLC). OT extends beyond control systems to include directory structures and associated computing infrastructure, communications networking systems and specific software.
Pipelines	A cloud-based, automated service that builds, tests, and deploys code projects, as part of the Azure DevOps suite
Programmable Logic Controller (PLC)	PLC is a digital computer used for automation of electromechanical process such as control of machinery on assembly lines or in plants (e.g. water treatment plants)

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	19 of 21

Term	Definitions
Project	The overall goal that is being achieved. The project encompasses all the requirements that are created or the overall solution that a development team may come up with.
Requirements/Changes	These can include new applications or functionalities that need to be produced, or it could include a simple fix to how an existing solution works.
Software	Refers to code produced/created for Seqwater. Examples include: • PLC code • Infrastructure as Code. Software can be seen as a moving part within a larger IT System
Solution	Refers to the code that is produced by the development team.
Supervisory Control and Data Acquisition (SCADA)	The main function of SCADA is for acquiring data from remote devices such as valves, pumps, transmitters etc. and providing overall control remotely from a SCADA host software platform. SCADA host platforms also provide functions for graphical displays, alarming, trending, and historical storage of data.
SVN	An abbreviation for Subversion, which is an open-source version control system.
System	Refers to the entire set of applications, software and infrastructure that supports the IT operations of Seqwater.
Worker(s)	Includes all permanent, temporary, and casual employees of Seqwater, and: • vocational and work experience placements • volunteers • contractors and consultants employed by another entity but temporarily assigned to do work for or on behalf of Seqwater.

6. References and Related Materials

Description	Location
D24/206099 Change Management Information Sheet - Mandatory Deliverables for Normal Changes	REX
SPE-00531 Cyber Security Assurance Specification	REX
Guide to the Software Engineering Body of Knowledge (SWEBOK), V4.0	
I-REG-STD-002 PLC Std Block Data sheet and Test Reports Document Register I-REG-STD-004 SCADA Std Objects Data Sheet and Test Reports Register	Waternet & REX
ISO/IEC 12207 – Software development lifecycles https://waternet.corporate.local/AssetMan/Engineeringservices/Pages/Australian-Standardsd-Web-Service.aspx	Waternet

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	20 of 21

This document is the property of Seqwater. It must not be copied or reproduced in any way whatsoever without the authority of Seqwater. This document is uncontrolled when printed. An electronic database manages and stores the controlled version.

Description	Location
PLN-00388 Monitoring and Control Systems Asset Class Plan	REX
POL-00091 Information Security Management Policy	REX
POL-00092 Technology Policy	REX
POL-00116 Enterprise Architecture Policy	REX
POL-00118 ICT Change Management Policy	REX
POL-00119 ICT Asset Management Policy	REX
POL-00160 Citizen Development Management Policy	REX
PRO-01816 ICT Change Management Procedure	REX
PRO-01874 Engineering Standard X-PRO-STD-004 Development and Review of Asset Standards Procedure	REX
PRO-02205 Engineering Standard X-PRO-STD-008 Asset Standards Management and Application Procedure	REX
PRO-02234 Management of Change (Technical) Procedure	REX
PRO-02329 I-SPE-STD-019 Process Control Systems - Information Maintenance Procedure	REX
Seqwater Corporate Strategy 2024 – 2028	Waternet
SPE-00342 Engineering Standard I-LST-STD-001 Control Systems Prescribed Equipment	REX
SPE-00349 Engineering Standard B-SPE-STD-005 Functional Description Documentation	REX
SPE-00356 Engineering Standard I-SPE-STD-005 High Priority Alarms & Interlocks Management	REX
SPE-00357 Engineering Standard I-SPE-STD-006 Alarm and Event Management	REX
SPE-00359 Engineering Standard I-SPE-STD-010 Control Systems Tag Naming	REX
SPE-00360 Engineering Standard I-SPE-STD-011 Control Systems Testing	REX
SPE-00361 Engineering Standard I-SPE-STD-013 Control Systems Design and Construction	REX
TEM-00215 Engineering Standard X-TMP-STD-004 Request for Engineering Services Template	REX
TEM-00221 Engineering Standard I-TMP-STD-001 Functional Description Part A Template	REX
TEM-00222 Engineering Standard I-TMP-STD-005 Functional Description Part B Template	REX

Version No.	Document No.	Document Owner:	Approval Date:	Document Approver:	Page:
1	FRA-00050	Team Leader Process & Analysis	11/06/2025	Manager Service Delivery	21 of 21

This document is the property of Seqwater. It must not be copied or reproduced in any way whatsoever without the authority of Seqwater. This document is uncontrolled when printed. An electronic database manages and stores the controlled version.