ICT Project Guidance

System, Data, and User Experience   
Target Qualities

Author

: Sky Sigal, Solution Architect, TPHM | Infrastructure & Digital

## Synopsis

Stakeholders expect services they procure, develop, maintain, and support, and offer to meet a baseline of qualities.   
  
While the qualities they expect may at first appear arbitrary and case specific, ISO[[1]](#footnote-1) standards have been developed to define system, data and use experience qualities.

As per the *Business Analysis Body of Knowledge* (BABOK) v3.0, current best practice is to define desired qualities using terms defined in ISO standards for defining system, data and experience qualities.

Along with the development of and delivery to functional requirements, project delivery must allocate effort to the deliver and iterative improvement of these internationally defined and agreed qualities.

## Introduction

Qualities are used to develop Quality Requirements.

Prior to 2011, best practice was to develop the Requirements used for procuring ICT solutions as a set of Functional Requirements and Non-Functional Requirements.

Functional Requirements were biased to capturing Business User expectations, omitting other users. Non-Functional Requirements were used for everything else – often an messy combination of sparse functional requirements of non-business users (e.g., security, support, operations & maintenance), some cross-functional and system functional requirements, desired system qualities, and possible also some transitional requirements.

Current best practice is an improvement.

## Quality Requirements

Current best practice is to develop Functional, Quality and Transitional requirements as 3 different sets of requirements.

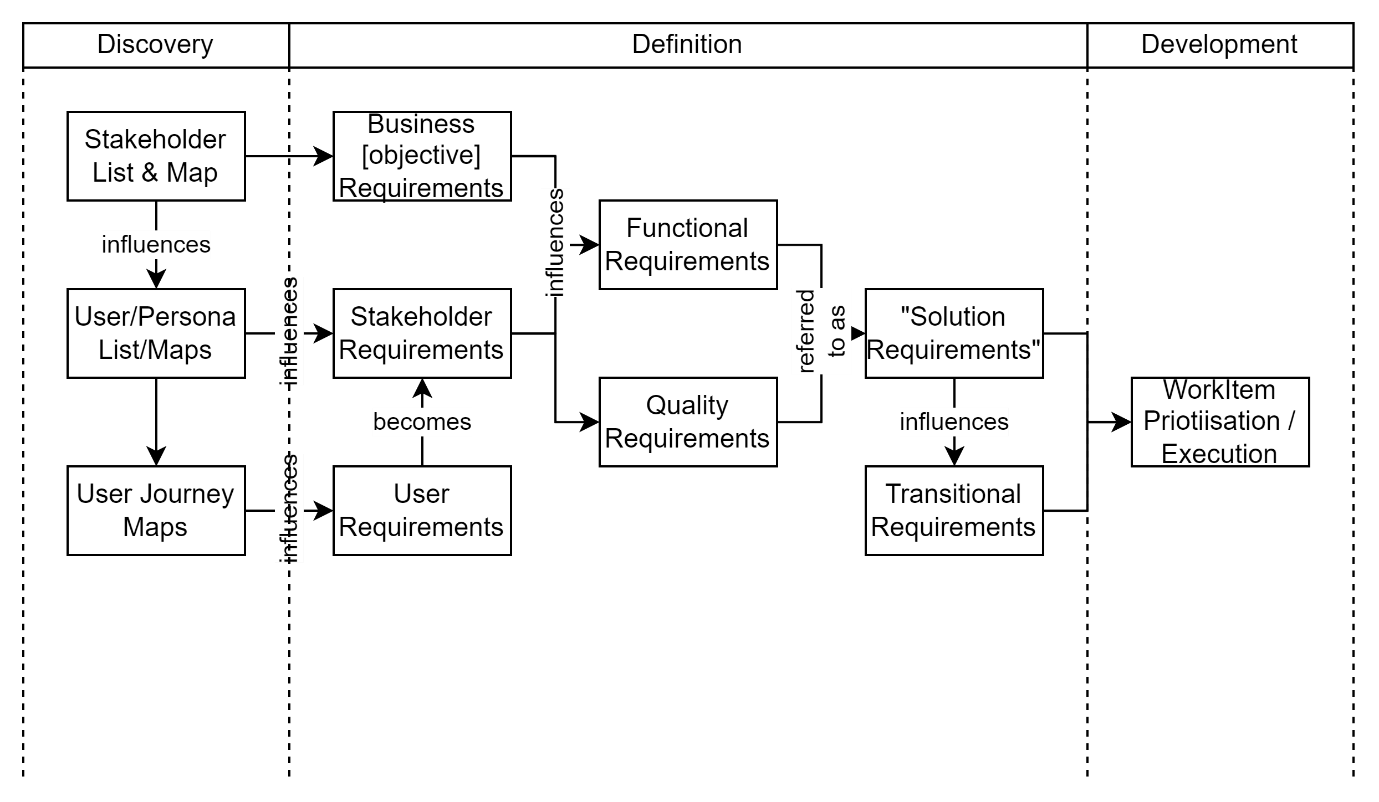
For Functional Requirements (FR), the recommendation is to start with a comprehensive stakeholder and user map to guide the gathering of functional requirements from business users right through to operations, maintenance, security, assurance and change management stakeholders.

For desired system qualities requirements (QRs), it is recommended to use ISO-25010,12,22 to develop a document that captures desires qualities.[[2]](#footnote-2)

Transitional requirements (TRs) are developed separately to both, and are used to capture data extraction, data quality, data and user provisioning, comms, etc. tasks required to transition from current state to target state.

## Quality Delivery

The developed requirements are used for contractual purposes, and later for developing and prioritising the work items (User Stories and Tasks) required to deliver the contracted work.



## Quality Assurance

Most qualities are binary qualitative in nature (they are either done or not) and are not quantitative in nature.

## Assuring Qualitative Qualities

Qualitative Qualities can be validated by direct sighting or indirect sighting of a report generated by automation of quality tests in the pipeline. Either approach can be used to ensure qualities such as: the site is discoverable at a recognisable URL, requests to insecure (HTTP) endpoints are redirected to secure (HTTPS) equivalent endpoints, etc.

## Assuring Quantitative Qualities

A very small number of quality requirements are required to be quantitative, in that they specify target availability thresholds.

These quantitative requirements are often intertwined. Examples include:

* + - * the maximum synchronous duration of most operation (e.g., “90% of requests will be defined as Standard requests, returning control to users in less than 0.5 seconds, 8% will return control in less than 2 seconds, and 0% will take more than 15 seconds to complete and return control to user”).
      * the number of “standard” requests that can be processed at the same time, while meeting infrastructure thresholds
      * systems CPU usage will remain below 33% while handling standard requests, peaking above for no more than 5 seconds.

The above quantitative qualities are often more expensive to validate, requiring setting up performance rigs for example.

# Qualities

For developing Quality Requirements, three key ISOs have been made available to guide the design, development, delivery, operations, and maintenance of systems.

These ISOs are:

Table : List of ISOs that define ICT relevant Qualities

|  |
| --- |
| ISO-25022 : End user expectations of Qualities experienced from using the system. Thes experience qualities in turn rely on the qualities of the data they can access (ISO-25012) and the system that hosts it (ISO-25010), described next. |
| ISO-25012 : the qualities of the data the system manages, which in turn relies on the system’s qualities (see ISO-25010). |
| ISO-25010 : the qualities of the system itself, within which the data (see ISO-25012) is kept. |

This document summarises the different qualities described in each list, along with relevant and achievable recommendations to deliver to these quality expectations.

## Application

Delivery managers should request that when work items (User Stories, Tasks) are developed they are categorised by what Quality they contribute to. This enables reporting as to what qualities have been delivered, and later, permits a Quality Analyst to quickly determine what to test to ensure the related Quality has been delivered to delivered to expectations.

In essence the categorisation and measurements provide Project Delivery Managers insight as to where to apply effort to iteratively improve qualities alongside delivering functionality.

## End User Quality Expectations

ISO-25022 defines the Qualities expected by End Users of a Service in Use.



Figure : ISO-25024 Qualities in Use (HL)

While the ISOs describe a means to measure the above, the Qualities are for the most part subjective. But while subjective or measured it is important to note that rely on the delivery of less subjective qualities defined next (ISO-25010, ISO-25012, etc).

Note: subjective readings can be collected actively, via short questionnaires, or passively, by measuring other available metrics (likes divided by sign ins, usage areas, etc.).

Table : ISO-25022 Qualities In Use (HL)

|  |
| --- |
| Effectiveness An iteratively improvable subjective metric of the system’s ability to improve the effectivity of a User operations.   Contributors to this outcome may include:   * + - * Changes to system data effects change, directly (by messaging other systems of the changes), or indirectly, by producing a report that can be relied upon to make decisions.       * Automated workflows based on form inputs, that reduce workloads. |
| Efficiency An iteratively improvable subjective metric of the system’s ability to affect the efficiency by which a User can perform their tasks to completion.  Contributors to this quality may include:   * + - * the number of steps required to provide information or instructions,       * time required to return to user input readiness state (see ISO-25010/Usability) and time it took for the operations to complete (e.g.: asynchronously). |
| Satisfaction An iteratively improvable subjective metric as to how much satisfaction a user derived from performing tasks with the system. |
| Contributors to this quality may include:   * + - * signalling that time spent affected a desired change.  By reporting completion of tasks and producing a means of viewing the change by graphs, charts, and/or reports.  Safety An iteratively improvable subjective metric as to how safe a User feels operating the system (e.g.: doesn’t expect it to lose data, or if an error is done, that it’s not unrecoverable, or that it sends out notifications to the wrong person).  Contributors to this quality may include:   * + - * having a means to inform users of tracking, data usage and storage purpose and policies etc.       * having the means to signal poor value data, poor value behaviour, poor value outcomes, with an understanding of what will happen next, potentially describing clearly who, how, how long investigation and potential resolution should take.   Note: Safety is not the same as Security. |
| Context Comprehensive An iteratively improvable subjective metric as to the comprehensiveness of the service offered by a system.   Contributors to this quality may include:   * + - * ISO-25010/Functionality/Completeness metric,       * And/or reflecting the number of items in the backlog for certain domains of functionality, |

## System Information Quality Expectations

ISO-25012 defines the Qualities expected by end users of the Information made available via the Service.

ISO-25012 is notable in that it covers both the qualities of data itself, and the qualities of the system on which these data qualities rely upon.

Note:   
the system qualities in ISO-25012 are ones that are also included within ISO-25010 (System Qualities), described next.

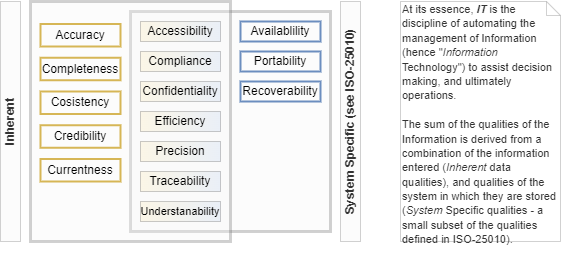


Figure : ISO-25012 System Information Qualities (HL)

Table : ISO-25012 System Information Qualities

### Data Qualities

The first set of qualities is specific to the Data itself:

|  |
| --- |
| Accuracy A metric as to the accuracy of the data, improvable by validation & automated correction.  Contributors to this quality may include:   * + - * the Availability of a system, which in turn does not delay service clients from sending updated information regularly, potentially often, potentially in multiple operations,       * The availability of authenticated APIs API for service clients to send updated information without requiring an end user to use screens,       * The auditing of changes, and values provided, linkable to source, to contact systems that provide non-accurate information, and request they improve their data.       * Defining an expected frequency of updates which in turn improves the currency of the data.       * Considering restricting information input to trusted credible sources,       * The validating submitted info against authoritative sources.   The system managed data is expected to be Current, therefore Accurate to a Usable level, appropriate for making Decisions that can be used to improve learner outcomes. |
| Completeness A metric as to the completeness of the data, improvable by auto completion, validation, workflow design, etc.  Contributors to this quality may include:   * + - * The breadth of data the system manages, by developing a target list of data types of interest, and managing the completion of the functionality to store, validate and manage them,       * The depth of the data the system manages, by listing what metadata (author, when, datatype, source filename, etc.), categorisation (tags/folders) and linkages (relationships, sequencing, etc.) in between the elements to store, validate and manage,       * *The system’s purpose and therefore data regarding all providers in NZ, learners, their associated longitudinal data and their current enrolments is expected to be complete over time.* |
| Consistency A metric to the consistency of data across multiple services, including service consumers.  Contributors to this quality may include:   * + - * The validation of data against known rules and/or systems of record known data,       * The auditing of change,       * The potential tracking of multiple ids per data item, permitting the asynchronous signalling back to source system of inconsistencies.   *The data in the system is update by and therefore consistent with Education Sector Registries.*  *Further data integrations from Education Providers are compared with this baseline before accepting further updates.* |
| Credibility A subjective metric as to the credibility of the source of the data. Improvable by authentication, the transparency of auditing, and integration sources chosen.  Contributors to this quality may include:   * + - * The use of the domain names that start with the top-level domain of the organisation (e.g.: *myservice.myorg.tld*),       * The use of organisation logos and recognisable media,       * The sending of notifications from an email system/other that can be associated to the organisation (e.g.: *notifications.myorg.tld*),       * The system is hosted on infrastructure that is managed by the organisation,       * Etc.   The data is hosted on a system managed & actively promoted by the Ministry of Education. |

### Data and System Qualities

The above data qualities in turn rely on qualities that are specific to how the data is made available by the system:

|  |
| --- |
| Currentness A metric as to the currency of the information. Improvable by rate of refreshment of data from authoritative sources, periodic prompting users to validate their information, etc.  Contributors to this quality may include:   * + - * The system’s availability permitting both synchronous, asynchronous updates       * Tracking updaters, so as to better report on the currency of data due to the data source.  Accessibility A metric as to the accessibility of the service’s information.  Improvable by ensuring information is available via standards based accessible, multi-language, multi-cultural interfaces.  Contributors to this quality may include:   * + - * The system remains available by being hosted on dynamically & horizontally scalable infrastructure,       * The system media (screen text & images) resources are managed separately to display, making it easier to have translated and accessible and rendered in culture and language-specific UI interfaces.       * User Profiles of some kind are used to store user preferences for the language in which to display resources – and optionally screen colour schemas, layout, etc.       * The user interfaces are regularly tested (preferably in an automated pipeline) for Web and Accessibility Standards.   Note: English based baseline system testing is a concern of CoreFour. Custom and/or Māori based interfaces should be decided whether CoreFour is an acceptable agent to act on behalf of this project, or whether additional independent QA testing is required by this project’s stakeholders. |
| Compliance A metric as to the compliance of the information.  Contributors to this quality may include:   * + - * The ability to meet national sanctions by being able to exclude the use of the service to requests originating in certain countries/regimes,       * The ability to limit access to secure communication channels (e.g.: HTTPS)       * The ability to audit all session Operations, whether by authenticated and public users, including Views,       * The storing of data in universal formats when they are available (e.g.: UTC DateTime) translated for locality when displayed (e.g.: local date/time format and zone).       * The use of universal Id formats when available (UUIDs)       * The system logically deletes data that is to be removed from access and circulation upon request (as per EE’s privacy policies) and after a configurable duration after the data is no longer needed for the stated business function.       * The system is capable of both deferring and later removing the use of reference data and relationships (e.g.: Group and Role Allocations) by using start and end date metadata and logic.   The storage schema of the SaaS system is proprietary to the Service Vendor, but the schema of the Interoperability APIs is designed by the solution’s data architect to meet enterprise standards and obligations. By extension it implies the internal schema is capable of being compliant. |
| Confidentiality A metric as to the confidentiality of the information.  Contributors to this quality may include:   * + - * Appropriate Access to the system managed information is limited to Authenticated & Authorised system Principals (Users).       * Using only secure channels for communication (e.g., HTTPS)       * Ensuring confidential information is only transmitted over secure channels (e.g., cookies are marked Secure and HttpOnly, and unencrypted requests are redirected by the server to the secure equivalent endpoint, etc.)       * Session Operations are logged for later auditing to ensure others have not Viewed or edited the record in question. |
| Efficiency A metric as to the efficiency of the information processing aspect. Improvable by design decisions to limit duplication of user and system effort to input or retrieve information. Improvable by ensuring data is only deleted logically and/or versioned, to provide an efficient means to restore information after input error.  Regular improvements to cloud based hosting hardware, and software efficiency improve the efficiency by which data is managed.  Contributors to this quality may include:   * + - * Prefilling of field values based on previously determined information,       * Reduction of double handling,       * Removal of unnecessary screen flow stopping modal views asking for confirmation before actions (prefer making the system able to Undo changes). |
| Precision A metric as to the ability to store information without loss of precision. Contributors to this quality may include:   * + - * validation of API (and therefore UX if API-First) based requests, arguments given, data schema types used,       * comparison of given data against records retrieved from systems of record. |
| Traceability A metric as to the ability to determine the source of changes to information.  Contributors to this quality may include:   * + - * ensuring auditing is enabled for all information change operations, traceable to the Session, and by that the User if Authenticated,       * recommending to doing the same for all viewing operations,       * ensuring that auditing metadata is not removed if the record is deleted,       * and strongly recommend that data is never physically deleted, only logically removed from use and availability.       * That changes do not replace records but add a new versioned without replacing the previous version.       * These last two recommendations are the means– beyond Usability advantages – by which records can demonstrate the current state based on all previous changes |
| Understandability A metric as to the understandability of the information.  Contributors to this quality may include:   * + - * signalling to users that operations were completed as expected, preferably unobtrusively and asynchronously.       * Signalling how to correct issues if they require intervention.       * Improving interoperability by developing APIs view schemas understandable to a system’s user base, while hiding the internals flexibility by abstraction. |

### System Qualities

The above defined data and presentation qualities rely in turn on the system having the following qualities:

|  |
| --- |
| Availability A metric as to the availability of the system.  Contributors to this quality may include:   * + - * Hosting on services with a provably high availability themselves,       * When relying on dependency services use ones with a high availability ratio, as well as providing secondary rollover services,       * Having in place a means, on another site, to signal to users when a system is planned to be down, and if unexpectantly, the current status, what RTO to expect, and what the RPO is.       * Having in place a backup that reduces the RPO.   Note: The exact times for RPO & RTO are defined in the project’s Quality Requirements, matching or exceeding other services managed by the organisation.  The Azure infrastructure on which the system is built has a measured average overall availability of 99.96% in 2022 – with higher numbers (up to 99.99%, financially backed) for specific services. |
| Portability A metric as to the ability of the information be used in other systems. Improvable by using standards for storage type, encoding, etc.  Contributors to this quality may include:   * + - * A data schema and internal that permits different accounts being in the same database without requiring different databases or tenancy ids (both dated design approach).       * An API to extract all information for one’s account.   Data is currently extractable via at least three different approaches:   * Standards based proprietary APIs (See Interoperability View), where data is formatted using international encodings (UTF-8) and Standards (e.g.: Dates, Times, Country Codes, etc.). * via EdsbyLink if so desired, transformed to meet individual target system requirements, standards based or not. * via standards-based NZ-SIF API, intended for use by SMSs as well as other systems upon their requesting it. |
| Recoverability A metric as to the ability to recover previous versions of information.  Contributors to this quality may include:   * Improvable by implementing Versioning, * Using Logical Deletion only, * Doing periodic full and incremental backups, * permitting end user-initiated restoration and roll backs to only ones account of information. |

## Information System Quality Expectations

Stakeholders desire systems – including Information systems as Edsby is an example of - to meet expected Qualities as defined by ISO-25010.

Note: ISO-25010 replaces the earlier and legacy FURPS approach to qualifying systems.



Figure : ISO-25010 System Qualities (HL)

Note also that ISO-25010, although still the most current standard used to define desired qualities of systems, was developed over a decade ago, prior to Privacy becoming a first-class concern in the IT industry.

Table : ISO-25010 System Qualities (HL)

|  |
| --- |
| Security An iteratively[[3]](#footnote-3) improvable metric of the security of the service – inclusive of the infrastructure and system code deployed to it.  Contributors to this quality may include:   * HR hiring controls, * Using deployment pipelines to limit access to environments and data, * deploying to ISO-27001 compliant hosting environments, * relying on static code analysis testing of source code to find weaknesses, * using independent security testing when changing security controls designs, * restricting and auditing access to production environment infrastructure, system, data and information. * Security metrics are expected to be collected primarily from 3rd party Security QA reports. * The SaaS and integration components are hosted on Azure, an ISO-27001 certified environment, the infrastructure and system is subject to regular vulnerability scanning and independent security testing, data is secure at rest (data encrypted storage within ISO-27001 compliant environments), and in transit (current SLA 1.3+ based encryption). |
| Privacy An iteratively[[4]](#footnote-4) improvable metric of the Privacy of the service.  Contributors to this quality may include:   * limiting the amount and scope of Personally Identifiable or Confidential information collected, * obtaining consent first and informing users as to what information is collected, for what purpose, for how long stored, and how its quality can be improved and/or corrected, * ensuring that Personal Information is only made available to Appropriately Authorised Users and Service Clients. |
| Operability An iteratively improvable metric of the system’s Functionality being operable by intended System Users.  Contributors to this quality may include:   * improvements to system flow, * improvements to user interface design to reduce barriers to use, * improving accessibility by adhering WCAG recommendations and meeting WCAG level definitions, * improving accessibility by permitting change of communication language and media used * designing user interfaces to be used by the youngest target audience cohort, or permitting the user interface to be variable for different cohorts.   The service’s User Interfaces are intended for and designed for use by Intermediate Learners upwards their Whānau and Parents, and providers staff (Teachers, Provider Administrators. Principals) and invited Sector/Ministry specialists. |
| Usability An iteratively[[5]](#footnote-5) improvable metric of the service’s user interface ability to be Recognisable and actionable by intended System Users.  Contributors to this quality may include:   * Improvable by providing wider access to more user roles, more functionality, while meeting usability and accessibility interface design standards, * Usability metrics can be sighted, baseline tested by specialists, and optionally ongoing automated tests using standard industry tools. The system’s UI are designed to be usable by Intermediate Provider users without undue assistance from other Learners, Parents, Whānau, Teachers. The system’s UIs are currently designed to be usable by Whanau members of medium IT capabilities. The system has been tested by QA specialists to meet NZ’s legal obligations prior to its initial deployment. |
| Functionality An iteratively improvable metric as to completeness and usefulness of the functionality provided by the service.  Contributors to this quality may include:   * increasing the outcomes delivered by functionality, not necessarily by increasing the scope of functionality offered. The functionality provided is always improvable, but currently sufficient -- as demonstrated by the large number of stakeholder roles and countries currently using the service. |
| Performance An iteratively improvable set of metrics of the responsiveness and throughput of the service.  Contributors to this quality may include:   * infrastructure and design choices to increase availability, caching, reduce latency, processing effort and data transfer between tiers, * The use of Performance metrics can be periodically collected by logging operational metrics (average execution duration, etc.) to automate dynamic changes in resource allocation (e.g. horizontal scaling),   Performance impacts by the technology choices of the vendor and the lag between NZ and AU cloud hosting are less than minor. Note that the distance lag is improvable when the cloud hosting provider opens a sovereign data centre on NZ territory and the service moved to it. |
| Maintainability An iteratively improvable metric as to the quality of the system’s maintainability in terms of code, data schemas, etc.  Contributors to this quality may include:   * Improving code maintainability by following design patterns to reduce distracting code development effort (by incremental implementation of automation of repetitive tasks, such as deployment), * improving certainty of results (by following TDD patterns and automating security checks), * designing to reduce complexity (by following DDD system component design patterns), * following Object Oriented (OO) development patterns to reduce coupling while increasing cohesion of the code   Maintainability can be measured by sighting and to a certain extent by static analysis test automation. While the core SaaS components maintained as part of the Edsby service -- and not an externally measurable metric – the maintainability of the code for integration and support component custom development is measurable via standard industry static analysis tooling. |
| Reliability An iteratively improvable metric of availability of the service.  Contributors to this quality may include:   * Using reliable hosting infrastructure service, probably cloud, certainly ISO-27001 compliant, * ensuring automated backups & restoration processes, * has full and incremental backup at a high cadence, * automated and periodic backups are for *all* mutable data storage used: relational, nosql, potentially including redis cache servers elastic storage, * potentially providing active-active redundancy (active passive is a deprecated pattern that just wastes 50% of infrastructure costs), * periodic testing of data restorations is expected, preferably automated, * The use of backup restoration can be tested via automation, * Data, If using cloud infrastructure, is in a different data zone than logic servers, * Data is restorable to a specific time in the past 31 days     The service’s system components are hosted on the infrastructure of the second largest cloud service provider -- the same provider used by most if not all Government Agencies in NZ – with uptime ratio within contracted defined and accepted limits. The techniques, backups cadence, are different depending on whether for SaaS or organisation hosted components.  While we are not responsible for Edsby service backups, it provides assurance that their four distinct data storage services (SQL Server, Redis, Elastic and Blob) use both duplication and backup strategies.  SQL Server data is both duplicated to different geographic zones as well as backed up to permit rolling back to any time within the past 35 days. Redis is backed up every 5 minutes to Blob storage. Blob storage is restorable to any specific time in the past 35 days. Elastic is not backed up, because it is rebuildable dynamically.  Ministry managed services use Ministry architecture patterns (incremental backups <=15minutes, full backups <=every day) to meet standard organisation RPO and RTO targets. In fact, we use the same standard Azure supplied 30+day, point in time processes as CoreFour are using for SQL Server based data. |
| Compatibility An iteratively improvable metric of the ability for the system to export data to external services.  Contributors to this quality may include:   * improvable by using standards for Integration, Transport, Encoding, and data interchange, * although avoid relying on legacy standards when more modern and lighter patterns are available and in common use (e.g., avoid SOA & SOAP based patterns, when REST is available).   The use of standards can be sighted and/or tested via automated tests.  The information within the system is made available to other systems via standards-based communication protocols (JSON over HTTPS) over standard development patterns (REST).  Additionally, most pertinent information is available upon request from NZ-SIF compliant APIs. |

Appendices

Appendix A – Document Information

### Diagrams

[Figure 24: ISO-25024 Qualities in Use (HL) 4](#_Toc145067031)

[Figure 25: ISO-25012 System Information Qualities (HL) 7](#_Toc145067032)

[Figure 26: ISO-25010 System Qualities (HL) 17](#_Toc145067033)

### Tables

[Table 1: List of ISOs that define ICT relevant Qualities 4](#_Toc145075280)

[Table 2: ISO-25022 Qualities In Use (HL) 5](#_Toc145075281)

[Table 3: ISO-25012 System Information Qualities 7](#_Toc145075282)

[Table 4: ISO-25010 System Qualities (HL) 14](#_Toc145075283)

### Review Distribution

Distribution/Review List:

* + - * Sandy Britain, Enterprise Architect, TPHM
      * Amy Orr, Data Architect, TPHM
      * Dijana Sneath, Business Analyst, TPHM Analysis and Design
      * Archana Sahani, Business Analyst, TPHM Analysis and Design

## Conventions: Diagrams

Unless solely for a technical audience, where standards based industry diagram conventions (ArchiMate/UML/C4) are called for, diagrams are developed as simple “boxes & lines” for a broader audience.

### Terms

N/A

1. ISO-25010, ISO-25012, ISO-25022 [↑](#footnote-ref-1)
2. The ISOs provide a rigour of thought that was unavailable earlier in the era of using “FURPS”, “FURPS+”, etc. develop Non-Functional Requirements. [↑](#footnote-ref-2)
3. Note that legal obligations define a minimum required before a system can be deployed to a production data environment. [↑](#footnote-ref-3)
4. Note that legal obligations define a minimum required before a system can be deployed to a production data environment. [↑](#footnote-ref-4)
5. Note that legal obligations define a minimum required before a system can be deployed to a production data environment. [↑](#footnote-ref-5)