**Netcompany – Methodology and Security**

**A0130 - Requirement Specification**

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

|  |  |
| --- | --- |
| **Version:** | 2.1 |
| **Status:** | Approved |
| **Approver:** | Leni Støvring Barfred |
| **Author:** | Leni Støvring Barfred |



**Document history**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Date | Author | Status | Remarks |
| 1.0 | 18-09-2015 | Leni Barfred | Final |  |
| 2.0 | 04-05-2016 | Simon Shan Man Ng | Approved | 2016 New logo |
| 2.1 | 15-9-2020 | Leni Barfred | Approved | New colors |

References

|  |  |  |  |
| --- | --- | --- | --- |
| Reference | Title | Author | Version |
|  |  |  |  |

Table of contents

[1 Introduction 3](#_Toc51057885)

[1.1 Purpose 3](#_Toc51057886)

[1.2 Target audience 4](#_Toc51057887)

[1.3 Reference group 4](#_Toc51057888)

[1.4 Approval 4](#_Toc51057889)

[2 Guideline to A0130 5](#_Toc51057890)

[2.1 Description of the solution 5](#_Toc51057891)

[2.2 List of requirements 5](#_Toc51057892)

[3 Tools 6](#_Toc51057893)

[4 Recommendations from IEEE 830-1998 6](#_Toc51057894)

# Introduction

## Purpose

The requirement specification helps the customer to describe precisely what they want to achieve.

The requirement specification helps the supplier to understand precisely what the customer wants.

The requirement specification establishes a basis for an agreement between the customer and the supplier concerning what the solution will do.

The requirement specification reduces development effort. A careful review of the requirements permits identification of omissions, misunderstandings and inconsistency early on in the development progress, at a point when these problems are easier to rectify.

The requirement specification provides a basis for estimation of expenses and a schedule.

The requirement specification provides a basis for validation and verification.

A requirement specification will, in practice, be used for one or more of the following purposes:

* A general specification of the deliverable required by the customer. Including the elements that cannot be placed in any other deliverables from the analysis phase. This is typically relevant for many non-functional requirements. As a result, the deliverable may be an important starting point for the design phase.
* Providing guidance for the ambition level in respect of the scope of the project. Requirements are categorised into iterations. This can be used as a tool and act as a "lightning conductor" for any overambitious project participants.
* Providing guidance for prioritisation in connection with cuts in scope. Requirements should be prioritised mutually, particularly if the specification includes functional requirements.
* Appendix to a contract. Unambiguity and feasibility within budget therefore an important characteristic of the specification. A checklist to see whether all requirements are covered by design and testing deliverables and the implemented product respectively.

Before preparing the requirement specification, what the project needs from the deliverable should therefore be clarified. The question of what will be documented in the requirement specification and what will be documented elsewhere must be answered. Documenting the same thing in several places is inappropriate.

The requirement specification includes requirements for the solution (the deliverable). In the Netcompany method, we document requirements for the project in other deliverables. If it is necessary in any given situation to include formal requirements for the project in this deliverable – relating to organisation and schedules, for example – these must be kept separate from requirements for the deliverable.

A requirement specification is not necessarily the primary, authoritative and exhaustive starting point for the design phase (and hence the final resulting artefacts from a system development project). This often comes as a surprise to external project participants, who have no previous experience of system development. Therefore, you may meet resistance from a customer if the project decides that the deliverable is surplus to requirements or partially surplus to requirements in the context of the project.

Many Netcompany projects have been completed well without this deliverable. A number of projects have prepared a variant of the deliverable, according to the customer's expressed wishes where applicable, which was not used in reality by the project later (no value added).

If technical and formal requirements are specified in advance, it may be more efficient to use *A0140 – Functional Scenarios* and *D0160 - User-Interface Design* straight away to specify the scope of the project.

However, a requirement specification often has a reason to exist, particularly for documentation of requirements in non-functional areas; performance, architecture, standards, user-friendliness, documentation, etc.

## Target audience

In general, it must be possible for a requirement specification to be read by both technical and business stakeholders. The customer's and Netcompany's project managers must be able to understand the full significance of all requirements.

**Analysis phase participants**. The requirement specification is a way of establishing consensus among project participants concerning the scope of a project. The deliverable also often helps to give the customer's participants a sense of security. However, the product may also have the opposite effect due to a lack of transparency, so security (AKA "buy in") must be achieved in combination with use cases and page outlines, which are frequently more effective in this situation. The deliverable may also provide a significant starting point for estimation, so it must provide guidance on the effort required (ambition level and scope).

Preparation of requirement specification may form part of a **sales initiative**, either in competition with alternative suppliers or in competition for a budget. In this case, there must be particular emphasis on advantages.

The requirement specification may be an appendix to the **contract**. In this case, there must be particular emphasis on feasibility, accuracy and security.

The requirement specification will be one of the **starting points for preparation of the deliverables for the design phase**. It will also be necessary to verify the deliverables for the design phase against the requirement specification. Completeness is ideal for this purpose, but in practice it must not be pursued other than to ensure that the specification is accurate. A requirement specification is a general document and should not and will never be complete.

**Test phase participants**. The requirement specification will be one of the starting points for test preparation. It will also be necessary to verify the deliverables for the test phase against the requirement specification.

## Reference group

The deliverable must be prepared carefully with the customer's functional and technical experts, and stakeholders in general who may otherwise wish to get involved in the work of the project at an inopportune time. In principle the customer should designate stakeholders, but in fact Netcompany's project manager will end up with the most significant disadvantages unless the voices of all relevant stakeholders are heard.

Like the rest of the deliverables for the analysis phase, the requirement specification should ideally be characterised primarily by business requirements, not user requirements, and it should be characterised by planning and leading staff who are able to think ahead ("visionary"), and along the lines of alternatives to existing processes and workflows.

## Approval

The final version of the deliverable must be explicitly approved by one of the customer's project participants who is competent to agree on the scope of the project in a binding fashion. This is typically the customer's project manager.

The final version of the deliverable will be submitted to the project's steering committee for formal approval. A steering committee will typically merely base its approval on the recommendation of the customer's project manager.

# Guideline to A0130

The requirement specification may include the following elements:

* **A description of the solution** which describes in words and illustrations the functionality, context and business effect of the desired solution.
* **A list of requirements** which includes a list of explicitly formulated requirements for the solution.

The description of the solution may be surplus to requirements as the content is covered by other deliverables.

If Netcompany's job is limited to completing the analysis phase – as preparation for a tender, for example – a full requirement specification is typically an deliverable required.

## Description of the solution

The description must be kept general. It is more appropriate to provide a complete description of features and functionality in A0140 – Functional Scenarios.

At least the following topics should be considered:

* **Perspective for the solution**. The background and context for the project.
* **Features of the solution**. A general functional description. Do not repeat what appears in the list of requirements; rather, explain "why".
* **Non-functional requirements for the solution.** A general description. Do not repeat what appears in the list of requirements; rather, explain "why".
* **Stakeholders for the solution**
* **Technical platform for the solution**. Insofar as this is provided in advance.
* **Constraints for the solution**. Elements on which the design is dependent in context (what limits opportunities for the solution). This includes but is not limited to any external technical interfaces or subdeliverables from parallel projects.
* **Dependencies for the solution**. How the context is dependent on the solution (what limits the options of others, what is affected by changes to the external framework of the solution). This includes but is not limited to the effect of business plans or subdeliverables to parallel projects.
* Need for **documentation** of the solution.
* **Assumptions** made by the project as a basis for the description of the solution. As opposed to known facts. This section should often be used to specify explicitly what is not included in the solution.

## List of requirements

The list of requirements may include requirements in the following three categories:

* **Functional requirements**. For example, requirements relating to: features, user experience, information scope, functional performance, supported workflows, reports, processing online, batch processing …
* **Technical requirements**. For example, requirements relating to: standards applied and supported, technical performance, user-friendliness, accessibility, and other quality features, integration patterns, architecture used, use of specific tools, establishment of specific environments …
* **Formal requirements**. For example, requirements relating to: deadlines (should be avoided), documentation, licensing terms, dependencies and constraints relating to external conditions …

At least the following information is documented for each requirement (typically in columns):

* **ID**. Unique reference. Select numbering that does not refer to the contents of the requirement itself.
* **Description**. A description of what the requirement evolves. The description must be clear and the requirement must be feasible.
* **Priority**. Reflection of business value. The lower the priority, the greater the likelihood of the requirement being eliminated later due to budget concerns (or the requirement may be postponed until a later release). Requirements are often formulated by project participants who are not responsible for budgets, so it is easier to document a requirement as a low priority requirement for these people rather than discussing whether the requirement is relevant at all.
* **Iteration/Release**. Specification of the iteration in which the requirement will be met. This is also relevant in projects with just one planned release. A project budget is frequently provided for a number of iterations at a time. Requirements are often formulated by project participants who are not responsible for budgets, so it is easier to document a requirement as a "future" requirement for these people rather than discussing whether the requirement is relevant at all.
* **Degree of compliance**. Whether the project (project manager) decides to meet the requirement is documented. The degree of compliance may be indicated as "met", "partially met", "not met". If the degree of compliance is documented as "partially met", remarks must be attached indicating either the scope of compliance or a proviso.
* **Remarks**. A description is provided of how a requirement is met, where relevant, in fact often as a disguised proviso. Or possibly why a requirement is not met.

# Tools

The list of requirements is documented in the toolkit so that the requirements can be filtered and sorted. Use of Excel for the purpose has been seen but is not recommended☺ This is because the toolkit list may have history so that you can see who wrote what in a requirement, and the individual requirement may have a status so that you can control the process by describing the individual requirements more effectively.

# Recommendations from IEEE 830-1998

IEEE 830-1998 recommends that both the supplier and one or more of the customer's representatives should prepare the requirement specification. When a tender is compiled, the customer has typically already prepared a version.

The requirement specification must address the following topics

* Functionality
* External interfaces – to users, the system's infrastructure, external systems
* Performance – response times, uptime, capacity
* Architecture attributes – "-ilities" (e.g. maintainability)
* Implementation restrictions – whether there are standards in effect, programming language, database integrity, resource limits, operating system, etc.

Design or project requirements must be avoided when preparing the requirement specification. The requirement specification limits the range of valid designs but does not assign a specific design

A good requirement specification is

* Correct – every requirement in the requirement specification is a requirement which must be met by the solution
* Unambiguous – every requirement in the requirement specification has only one possible interpretation
* Complete – all essential requirements are included in the requirement specification
* Consistent – no requirements contradict one another
* Prioritised according to importance – not all requirements are equally important
* Verifiable – for every requirement there is a final cost-effective process where a person or machine can check that the requirement has been met. In general, ambiguous requirements are not verifiable (e.g. "works well", "good user interface"). If no method is established with regard to the meeting of the requirement, the requirement must be removed
* Modifiable – a requirement specification is modifiable if the structure is such that changes to requirements can be made easily, completely and consistently without changing the structure. Avoid redundancy. Express individual requirements separately and do not combine them with other requirements.

In the Netcompany method, however, a requirement specification should "only" be sufficient to meet the needs of the project.