Software requirements specification (SRS) - DRD

* 1. DRD identification
     1. Requirement identification and source document

The software requirements specification (SRS) document is called from the normative provisions summarized in Table D-1.

: SRS traceability to ECSS-E-ST-40 and ECSS-Q-ST-80 clauses

|  |  |  |
| --- | --- | --- |
| ECSS Standard | Clauses | DRD section |
| ECSS-E-ST-40 | 5.4.2.1 eo a., | <4.2>, <5.2>, <5.3>, <5.6> |
| b., | <5.5>, <5.2>, <5.9>, <5.11>, <5.12>, <5.13>, <5.14>, <5.17> |
| c., | <5.10> |
| d., | <5.8> |
| e., | <5.16> |
| f., | <5.15> |
| g., | <6> |
| i. | <5.7>b.7. |
| 5.4.2.2 | <5.7>b.5. |
| 5.4.2.3 eo a.,  b.,  c., | <8>  <8>  <8> |
| 5.8.3.2 eo a. | <7>, <5.1>c. |
| ECSS-Q-ST-80 | 6.3.2.4 | <5> |
| 7.1.1 eo b | <5.10> |
| 7.1.2 eo b | <5.10> |
| 7.2.1.1 eo b | <5.10> |
| 7.2.1.3 eo b | <6> |
| eo = Expected Output | | |

* + 1. Purpose and objective

The software requirements specification is a major constituent of the technical specification (TS). It describes the functional and non functional requirements applicable to the software item.

* 1. Expected response
     1. Scope and content

1. Introduction

The SRS shall contain a description of the purpose, objective, content and the reason prompting its preparation.

1. Applicable and reference documents

The SRS shall list the applicable and reference documents to support the generation of the document.

1. Terms, definitions and abbreviated terms

The SRS shall include any additional terms, definition or abbreviated terms used.

1. Software overview
   1. Function and purpose

The SRS shall describe the purpose of the product.

* 1. Environmental considerations

The SRS shall summarize:

1. the physical environment of the target system;
2. the hardware environment in the target system;
3. the operating environment in the target system;
   1. Relation to other systems

The SRS shall describe in detail the product’s relationship to other systems.

If the product is a component of an integrated HW–SW product, then the SRS shall:

1. summarize the essential characteristics of this larger product;
2. list the other HW or SW component the software interfaces with, and summarize the computer hardware and peripheral equipment to be used.

A block diagram may be presented showing the major components of the larger system or project, interconnections, and external interfaces.

* 1. Constraints

The SRS shall describe any items that limit the developer’s options for building the software.

The SRS should provide background information and seek to justify the constraints.

1. Requirements
   1. General
2. The following provisions apply to the software requirements listed in the SRS, as specified in <5.2> to <5.17> below.

Each requirement shall be uniquely identified.

When requirements are expressed as models, the supplier shall establish a way to assign identifiers within the model for sake of traceability.

The traceability information of each requirement derived from higher level documentation, to the applicable higher level requirement, shall be stated.

1. The documented trace can be provided automatically by tools when models are used to express requirements.

Requirements may be characterized, for example as essential or not, with a priority level to prepare incremental delivery, stable or not.

* 1. Functional requirements

The SRS shall describe the capabilities to be provided by the software item under definition.

The SRS shall provide where applicable the link between the requirements and the system states and modes.

Functional requirement shall be grouped by subject, in accordance with the logical model organization (e.g. per controlled subsystem).

Each requirement definition should be organized according to the following:

1. General
2. Inputs
3. Outputs
4. Processing

The SRS shall describe the functional requirements related to software safety and dependability.

* 1. Performance requirements

The SRS shall list any specific requirement to the specified performance of software item under definition.

* 1. Interface requirements

The SRS shall list and describe (or reference in the ICD) the software item external interfaces.

The following interfaces shall be fully described either in the SRS itself or by reference to the ICD:

1. interfaces between the software item and other software items;
2. interfaces between the software item and hardware products;
3. interfaces requirements relating to the man–machine interaction.

Naming convention applicable to the data and command interface shall be also described.

The definition of each interface shall include at least the provided service, the description (name, type, dimension), the range and the initial value.

* 1. Operational requirements

The SRS shall list any specific requirement related to the operation of the software in its intended environment.

The information specified in <5.5>a. should include, at least, any specified operational mode and mode transition for the software, and, in case of man–machine interaction, the intended use scenarios.

Diagrams may be used to show the intended operations and related modes–transitions.

* 1. Resources requirements

The SRS shall describe all the resource requirements related to the software and the hardware requirements (target hardware on which the software is specified to operate), as follows:

1. List of the requirements relevant to hardware environment in which the software is specified to operate.
2. List of the sizing and timing requirements applicable to the software item under specification.
3. Description of the computer software to be used with the software under specification or incorporated into the software item (e.g. operating system and software items to be reused).
4. Description of the real time constraints to respect (e.g. time management with respect to the handling of input data before its loss of validity).
   1. Design requirements and implementation constraints

The SRS shall list any requirements driving the design of the software item under specification and any identified implementation constraint.

Requirements applicable to the following items shall be included:

1. software standards (e.g. applicable coding standards, and development standards);
2. design requirements;
3. specific design methods to be applied to minimize the number of critical software components (see ECSS-Q-ST-80 6.2.2.4);
4. requirements relevant to numerical accuracy management;
5. design requirements relevant to the “in–flight modification” of the software item;
6. specific design requirements to be applied if the software is specified to be designed for intended reuse;
7. specific constraints induced by reused software (e.g. COTS, free software and open source).
   1. Security and privacy requirements

The SRS shall describe any security and privacy requirement applicable to the software item.

* 1. Portability requirements

The SRS shall list any portability requirement applicable to the software item.

* 1. Software quality requirements

The SRS shall list any quality requirement applicable to the software item.

* 1. Software reliability requirements

The SRS shall list any reliability requirement applicable to the software item.

* 1. Software maintainability requirements

The SRS shall list any maintainability requirement applicable to the software item.

* 1. Software safety requirements

The SRS shall list any safety requirement applicable to the software item.

* 1. Software configuration and delivery requirements

The SRS shall list any requirement applicable to the selected delivery medium and any software configuration applicable to the software item.

* 1. Data definition and database requirements

The SRS shall list any requirement related to specific data format or structure to be exchanged with other systems or any database requirements allowing to take into account e.g. for a flight software, the mission and product specific constraints.

* 1. Human factors related requirements

The SRS shall list any requirement applicable to:

1. the personnel and to the specific software product under definition;
2. manual operations, human–equipment interactions, constraints on personnel, concentrated human attention areas and that are sensitive to human errors and training, and human factors engineering.
   1. Adaptation and installation requirements

This SRS shall list any requirement applicable to adaptation data and to specific installation.

1. Validation requirements

The SRS shall describe, per each uniquely identified requirement in <5>, the validation approach.

A validation matrix (requirements to validation approach correlation table) shall be utilized to describe the validation approach applicable to each requirement.

1. Traceability

The SRS shall report the traceability matrices

1. from the upper level specification requirements to the requirements contained in <5> (forward traceability table), and
2. from the requirements contained in <5> to the upper level applicable specification (backward traceability table).

In case the information in <7>a. is separately provided in the DJF, reference to this documentation shall be clearly stated.

1. Logical model description

The SRS shall include a top–down description of the logical model of the software.

1. 1 The logical model can be the result of an iterative verification process with the customer. It also supports the requirements capture, documents and formalizes the software requirements.
2. 2 A logical model is a representation of the technical specification, independent of the implementation, describing the functional behaviour of the software product. The logical model is written with a formalized language and it can be possibly executable. Formal methods can be used to prove properties of the logical model itself and therefore of the technical specification. The logical model allows in particular to verify that a technical specification is complete (i.e. by checking a software requirement exists for each logical model element), and consistent (because of the model checking).  
   The logical model can be completed by specific feasibility analyses such as benchmarks, in order to check the technical budgets (e.g. memory size and computer throughput). In case the modelling technique allows for it, preliminary automatic code generation can be used to define the contents of the software validation test specification.
3. 3 If software system co-engineering activities are considered, the logical model is a refinement of the following system models: data, application function, event and failure

The method used to express the logical model shall be described

Diagrams, tables, data flows and explanatory text may be included.

The functionality at each level should be described, to enable the reader to ’walkthrough’ e.g. the model level–by–level, function–by–function, and flow–by–flow.

The behavioural view of the software logical model shall be also described in the SRS.

1. This is particularly relevant for flight software applications.
   * 1. Special remarks

None.