Documents package for IEC 62304 Safety Class C

US, McKinney, TX +1 214 306 68 37 Finland, Vantaa +371 66 011 905 UAE, Fujairah +971 600 57 59 69 KSA, Riyadh +966 800 880 3035
www.scnsoft.com contact@scnsoft.com

Project charter

- [ ] The processes to be used in the development of the system.
- [ ] The deliverables (including documentation) of the activities and tasks.
- [ ] Software configuration and change management plan, including software of unknown provenance (SOUP) configuration items and software used to support development.
- [ ] Software problem resolution procedures for handling problems detected in the software, deliverables, and activities at each stage of the life cycle (i.e., during the development as well as post-release).
- [ ] Software development standards, methods, and tools.
- [ ] Procedure for identifying and categorizing software defects that a selected programming stack may introduce.
- [ ] Procedure for documenting evidence that demonstrates that these defects do not contribute to unacceptable risks.
- [ ] Software system and integration testing plan:
  - [ ] The required functionality of the software (as in the requirements specification and scope).
  - [ ] Implementation of risk control measures.
  - [ ] Specified functioning of internal and external interfaces.
  - [ ] Testing under abnormal conditions, including foreseeable misuse.
- [ ] Software unit verification plan:
  - [ ] Strategies, methods, and procedures for verifying software units.
  - [ ] Software unit acceptance criteria.
  - [ ] Does the software code comply with the requirements, including risk control measures?
  - [ ] Is the software code free from contradiction with the interface design of the software unit?
  - [ ] Does the software code conform to programming procedures or coding standards?

- [ ] If applicable:
  - [ ] Proper event sequence.
  - [ ] Data and control flow.
  - [ ] Planned resource allocation.
  - [ ] Fault handling (error definition, isolation, and recovery).
  - [ ] Initialization of variables.
  - [ ] Memory management and memory overflows.
  - [ ] Boundary conditions.
- [ ] Risk management plan.
- [ ] Documentation plan.
- [ ] Delivery procedure.

Requirements traceability matrix

Risk assessment matrix

Software requirements specification

- [ ] Functional and capability requirements:
  - [ ] Performance.
  - [ ] Physical characteristics (e.g., code language, platform, operating system).
  - [ ] Computing environment (e.g., hardware, memory size, processing unit, network infrastructure) under which the software is to perform.
  - [ ] Need for compatibility with upgrades or multiple SOUP or other device versions.
- [ ] Software inputs and outputs:
  - [ ] Data characteristics (e.g., numeric/alphanumeric data, data format).
  - [ ] Data ranges, limits, defaults.
- [ ] Software-driven alarms, warnings, and operator messages.

- [ ] Security requirements:
  - [ ] Those related to the compromise of sensitive information.
  - [ ] Authentication.
  - [ ] Authorization.
  - [ ] Security audit trail.
  - [ ] System security/malware protection.
- [ ] User interface requirements:
  - [ ] For manual operations.
  - [ ] For human-equipment interactions.
  - [ ] For tasks requiring focused human attention.
- [ ] Data definitions and database requirements.
- [ ] Installation and acceptance requirements of the delivered software at the operation and maintenance site(s).
- [ ] Requirements related to methods of operation and maintenance.
- [ ] User maintenance requirements.
- [ ] Regulatory requirements.

Software requirements verification report

Detailed software architecture

- [ ] Structure of the software.
- [ ] List of software items (DoxyGen).
- [ ] Interfaces between software items (DoxyGen).
- [ ] Interfaces between software items and external components/systems (DoxyGen).
- [ ] Functional and performance requirements of SOUP items that are necessary for its intended use.
- [ ] Segregation between software items that is essential for risk control, and assurance that the segregation is effective.

Architecture verification report

Detailed software design

- [ ] Specification of each software unit in sufficient detail to facilitate its implementation (DoxyGen).
- [ ] Detailed specification for interfaces (DoxyGen):
  - [ ] Between software units (DoxyGen).
  - [ ] Between software units and external components/systems (DoxyGen).

Detailed design verification report

Software verification report

Software validation report

List of residual anomalies

Residual anomalies verification report