
	SOP Integrated Software Development	PROTOCOL No.:
		SUPERSEDE NO.:
		DATE:

	SOP Integrated Software Development	PROTOCOL No.:
		SUPERSEDE NO.:
		DATE:

SOP Integrated Software Development

TITLE:	
AUTHORING GROUP:	
DATE:	
SUPERSEDE PROTOCOL NO.:	

Summary

This SOP describes how software as a medical device is developed. It integrates risk management and usability engineering activities into the process.

Process Owner <enter role of process owner>

Key Performance indicators <enter KPIs to be tracked for the Management Review>

General Notes

Integrated Process, Evolutionary Strategy

This process integrates risk management and usability evaluation activities into the Software development process. It therefore, covers requirements of IEC 62304, ISO 14971 and IEC 62366. There are no separate risk management and usability engineering processes.

The Software Development Process described in this SOP resembles an “evolutionary” strategy (IEC 62304:2006, Annex B), acknowledging that the user need is not fully understood and not all requirements are defined up front. Whenever requirements change, the preceding process steps and their outputs need to be re-done to ensure consistent and complete documentation.

Process Steps

1. Design Input

Based on business input and product ideas, the process for product certification and registration is initiated to create an initial device description (incl. medical device classification and software safety classification) and high-level vision for the planned product. Technical input is considered to assess whether the idea is feasible.

Business input could be:

- Conversations with prospective customers
- Market opportunities
- Internal ideas

Changes to the product also enter the process here (i.e., as a change request as defined in SOP Change Management).

Participants

CEO

CTO

CPO

Input	Output
Business input	Device Description
Technical input	Vision document
Product ideas	Medical Device and Software Safety Classification
Change Request	

2. Risk Management Planning

The risk management activities are planned and documented.

The Risk Management Plan defines criteria for risk acceptability in the form of a risk policy and a risk acceptance matrix. It defines risk acceptability both for individual risks and the overall residual risk. The risk acceptance matrix is created by performing the following steps:

- Estimate product usage over its lifetime
- Define categories for the severity of harm (for example, categories may range from zero harm to death of a patient)
- Define probability categories (for example, categories may range from “unthinkable” over “rare” to “certain”).
- Start by defining the least probable category which has an absolute occurrence number of less than one. From there on, the more frequently occurring categories are added with probability increments of 10^2 .
- Create the risk acceptance matrix and define which combinations of the categories are deemed acceptable. Use color coding: red combinations represent unacceptable risks; yellow combinations represent risks that are acceptable.
- Note: no fields are marked as green as all risks must be reduced as far as possible.

Participants

CPO

Subject matter experts, e.g. physicians

Input

Output

Device description Risk Management Plan incl. risk acceptance matrix

3. First Risk Assessment and Usability Evaluation Planning

In the first risk and usability assessment, a preliminary hazard analysis is conducted and an initial risk table is drafted.

Risk analysis is performed by conducting a Failure Mode and Effects Analysis (FMEA). This analysis includes the following activities:

- Identifying potential failure modes
- Identifying potential hazards, hazardous situations and harms in collaboration with subject matter experts (e.g. physicians)
- Estimating probabilities for the identified items and analyzing the severity of each harm, taking into account international standards, scientific studies, public reports, expert opinions and usability data.
- For software devices, the probability of occurrence of a hazard is assumed as 100%. The probability of each hazard leading to a hazardous situation and that leading to a harm must be estimated separately as part of the risk analysis. Multiplied, they present the overall probability per risk. In combination, overall probability of occurrence and the severity of harm are evaluated against the risk policy previously defined for the device.

If a risk is deemed unacceptable based on our Risk Policy, it may be mitigated through Risk Control Measures in the priority as listed below. In general, we try to reduce the severity and probability of risks as far as possible (AFAP).

1. Inherently safe design
2. Protective measures in the device or development process
3. Information for safety and/or training of users

Further, a usability evaluation plan is created which covers future formative and summative usability evaluation activities.

User needs with a focus on those related to hazards are specified. These will serve as input to the summative usability evaluation and are reviewed following the checklist for user needs review.

Participants

CPO

Subject matter experts, e.g. physicians

Input	Output
Device description	Preliminary Hazards Analysis
Risk Management Plan	Risk Table (Draft)
	Usability Evaluation Plan
	User Needs
	User Needs Review Checklist

4. Software Planning

Based on the device description, the user needs and the preliminary risk analysis, the next step is to plan software development by defining software requirements. These also include the user interface specification, e.g. wireframes, mockups or style guides.

A software development and maintenance plan is created following our template. Software versioning is to be specified in the plan and should typically follow semver, in a format: MAJOR.MINOR.PATCH. Significant changes will lead to major version changes and a change of the UDI-DI, while non-significant changes lead to minor version changes and changes of the UDI-PI only. Third-digit version changes ("patches") result from bug fixes (see SOP Change Management and SOP Certification and Registration).

The software system test plan is created based on the requirements. As requirements may change, the software system test plan is continuously updated to reflect those changes.

Participants

CTO

Software Engineer

Risk Manager

Usability Engineer

Input	Output
Device Description	Software Development and Maintenance Plan

Input	Output
Vision Document	Software Requirements incl. User Interface Specification
Change Request	Software System Test Plan
Risk Table (draft)	Risk Table (updated)
Preliminary Hazards Analysis	

5. First Review: Software Planning Review

Software requirements are reviewed by following the checklist for Software Requirements Review. If the review is successful, move forward to the next step. If it's not, the software requirements have to be reworked with possible changes to the risk analysis and user needs. In that case, move back to the relevant step above.

Participants

CTO

CPO

Product Manager

Output

6. Software Requirements Review (filled out)
7. Risk Table (draft)
8. User Needs

Subject Matter Experts, e.g. physicians

Software Architecture, Detailed Design and Implementation

Software architecture is created (and detailed design, if necessary). As the software development process follows agile methodology, the software architecture may change as new knowledge is gained during implementation. The end result should be that both the implementation and the documented software architecture are synchronized.

At a minimum, an architecture diagram showing all software systems including databases and networks is created. For each software system, public interfaces, are documented, e.g. REST APIs, internal methods.

SOUP is added/updated here, if necessary. For each SOUP, we specify the name, version, manufacturer, website link (incl. release notes and issue tracker), requirements and prerequisites. SOUP must be verified before moving to the next step. Possible SOUP verification criteria include sufficient test coverage by the author and being commonly used; correct SOUP functioning is also verified through software verification and software system testing in the following steps.

If new risks relating to software units and potential failure modes are discovered during this phase, they are added to the risk table.

Participants

CTO

Software Engineer

Input

Software Development and Maintenance Plan

Software Requirements

Software System Test Plan

Output

Implemented Software Items, i.e. code

Software Architecture (created/updated)

 Software Detailed Design
(created/updated)

SOUP list (created/updated)

Risk Table (updated)

7. S e

cond Review, Verification, Formative Usability Evaluation, Integration

The second review covers multiple activities:

- **Verification** of the software items based on code review and automated unit and integration tests
- **Formative Usability Evaluation** through a usability engineer whether the user interface has been implemented as specified

Code review is conducted based on the following criteria:

- Are all software requirements, software architecture and detailed design implemented correctly?
- Does the code adhere to coding guidelines which include requirements for documentation as specified in the Software Development and Maintenance Plan?

Upon successful verification, the implemented software requirement is integrated into the current code base as described in the Software Development and Maintenance Plan. The software units may be integrated only if all activities above were successful.

Participants

Software Engineer

Usability Engineer

Input

Implemented Software Unit(s) incl. User

Output

Code review result

Input
Interface

Output

Unit / Integration test result(s)
Formative Usability Evaluation
Assessment

8. Software System Testing

Based on the Software System Test Plan, software system tests covering all software requirements are performed.

If new risks are discovered during the system tests, they are added to the risk table.

If anomalies are encountered, they are added to the list of known anomalies and/or entered as new software requirements to be fixed.

Participants

Software Engineer

Input

Output

Software System Test Plan	Software System Test Protocols
	Software System Test Report
	Risk Table (updated)
	List of known anomalies (updated)

9. Validation / Summative Usability Evaluation

Validation is done as a summative usability evaluation.

A Usability Test is conducted in the context of the actual user needs in accordance with the Usability Evaluation Plan.

If new risks are discovered during the usability tests, they are added to the risk table.

Participants

CPO

Usability
eer

Users for
lity Test

Input
Output

User Needs

 Usability Test
 Tool(s)

 Labeling and Instructions for Use, if
 available

 Summative Evaluation
 Report

Usability Evaluation Plan

Risk Table (updated)

**10. Final Risk
 Assessment and
 Risk-Benefit
 Analysis**

The overall risk of the product is evaluated by analyzing all identified risks so far. If unacceptable risks exist, they are weighed against the benefits of the Medical Device as part of the Clinical Evaluation SOP and as specified by the Clinical Evaluation Report. We only continue to release the Medical Device if the benefits outweigh the risks.

If unacceptable risks remain which are not outweighed by the benefits, we consider adding new risk control measures and move back in to the relevant step in the process.

The finalization of the Risk Management Report is the prerequisite for finalizing the Software Safety Classification.

Participants

CEO

CTO

Input
Output

CPO

Preliminary Hazards Analysis

Risk Management Report

Risk Table

Software Safety Classification (final)

Clinical Evaluation

Software (Release Candidate)

11. Product Release and Labelling

Before release, it is ensured that all required process steps (software development, usability evaluation, risk analysis) have been completed. Release notes are created and the list of known anomalies is finalized. The software is only released if the remaining anomalies are deemed acceptable.

Finally, the software is assigned required labeling, including at minimum: * The (trade) name of the device and manufacturer address * The intended purpose of the device (where it is not obvious to the user) * An indication that the product is a medical device, following symbols and labels specified in ISO 15223-1 * CE marking * Unique Device Identifier (see product certification and registration process) * Software version number in accordance with the software development plan * Warning, precautions, contraindications and residual risks that need to be brought to the user's attention as outlined by risk control measures in the risk file.

Regulatory Release:

A product (version) is considered released with the release of its declaration of conformity or, in the case of a minor version update, with the release of its updated software release checklist. The documents have to be signed by both a member of Management and the Person Responsible for Regulatory Compliance (PRRC). The regulatory release is completed by following all steps of SOP for product certification and registration.

Technical Release:

Following the regulatory release of the device, market placement and release is carried out by following the deployment process.

Participants

CTO

PRRC

Input

Device Description
 Software Release Checklist
 Risk Analysis
 User Needs
 Software Requirements
 Software Architecture and Detailed Design
 Software Items incl. Verification
 Software System Test Report
 Usability Evaluation Results

Output

Released Software
 Software Release Checklist (filled out)
 Release Notes incl. list of known anomalies