UNI-REPM Safety Module – Complete Description

Change Control List

Date	Version	Changed Items/Chapters
2017-10-27	0.1	Creation of document

PART I. Motivation

Safety-critical systems consist of a set of hardware, software, process, data and people whose failure could result in accidents that cause damage to the environment, financial losses, injury to people and loss of lives [68][69].

In this context, the literature reports that software has collaborated to deaths and injuries in many safety incidents and safety-related catastrophes [68][70][71][72][73][74] and several studies have identified problems with the RE process of SCS [75][76][77][78]. Currently, software have been used to implement and/or control an increasing number of traditional as well as innovative functions that are made possible only by software [79]. Furthermore, software also handles functions that were controlled by humans [79].

Therefore, software is becoming a major source of hazards since it can give wrong instructions to system hardware, through actuators, that can lead to accidents and hurt people [79]. Hence, considering the relevance of maintaining high confidence in safety-critical software [80], a consensus in academia and industry is being established that safety concerns should be addressed early in the system lifecycle [68][69][79][81].

Organizations with high maturity levels tend to reduce requirements issues and make the system development process less challenge. However, requirements engineers need systematic guidance to consider the safety concerns early in the development process of a safety-critical system.

There are some RE assessment frameworks, for example, the Requirements Engineering Good Practice Guide (REGPG) [64], Requirement Engineering Process Maturity Model (REPM) [65] Market-Driven Requirements Engineering Process Maturity Model (MDREPM) [66], and others that allow organizations to evaluate the strengths and weaknesses [67] regarding the RE process.

However, these maturity models do not cover both market-driven and bespoke requirements engineering [62]. To fill this gap, the Unified Requirements Engineering Process Maturity model (Uni-REPM) was proposed but it does not consider the safety issues required for the development of a safety-critical system.

In this work, we propose a complete safety maturity module for Uni-REPM that organizations could use as a guideline to assure that they do not fall in the most common mistakes made by companies during the RE process of safety-critical systems.

Our goal is to provide an easier, understandable and secure way to organizations evaluate the maturity in key safety-RE process areas but also guide them to discover what they miss or need to achieve the maturity level they desire.

PART II. UNI-REPM Safety Module Overview

1. Introduction

Requirements engineering issues such as vague initial requirements, ambiguities in requirements specification, undefined requirements process, requirements growth, requirements traceability, and confusion between methods and tools [50][53][63] have a huge impact in the quality of a safety-critical system.

In this context, there is a consensus that the most cost-efficient place to correct many problems is in the RE phase [38], [92], [93]. Despite this, requirements engineering remains a neglected area [50][53][63] [83][84].

Requirements problems are less frequent in organizations with high maturity levels [82]. Therefore, the Uni-REPM safety module aims to reduce issues in RE during the development of safety-critical systems by addressing safety practices that should be covered in the RE process to reduce the gap between these areas.

In the next sections, we describe the module structure, sources of actions, its contents and how to use it to evaluate the maturity level of an organization.

2. Module Structure

The Uni-REPM safety module follows the dual-view-approach of Uni-REPM: Process Area view and a Maturity Level view.

The process area view allows to visualize the hierarchy of process that consists the model and faster discover practices of the same group. The maturity level view, on the other hand, defines sets of practices that compose a consistent and coherent RE process, and where the practices in one level supports each other as well as the more advanced practices on the next level [62].

The safety module follows the same hierarchy of Uni-REPM that defines three levels: Main process area (MPA), Sub-process area (SPA) and Action. Figure 1 presents the Safety module and its relationship with Uni-REPM. The module extends the Uni-REPM model by adding new SPAs highlighted in orange. Existing process outcomes were not altered and none were removed.

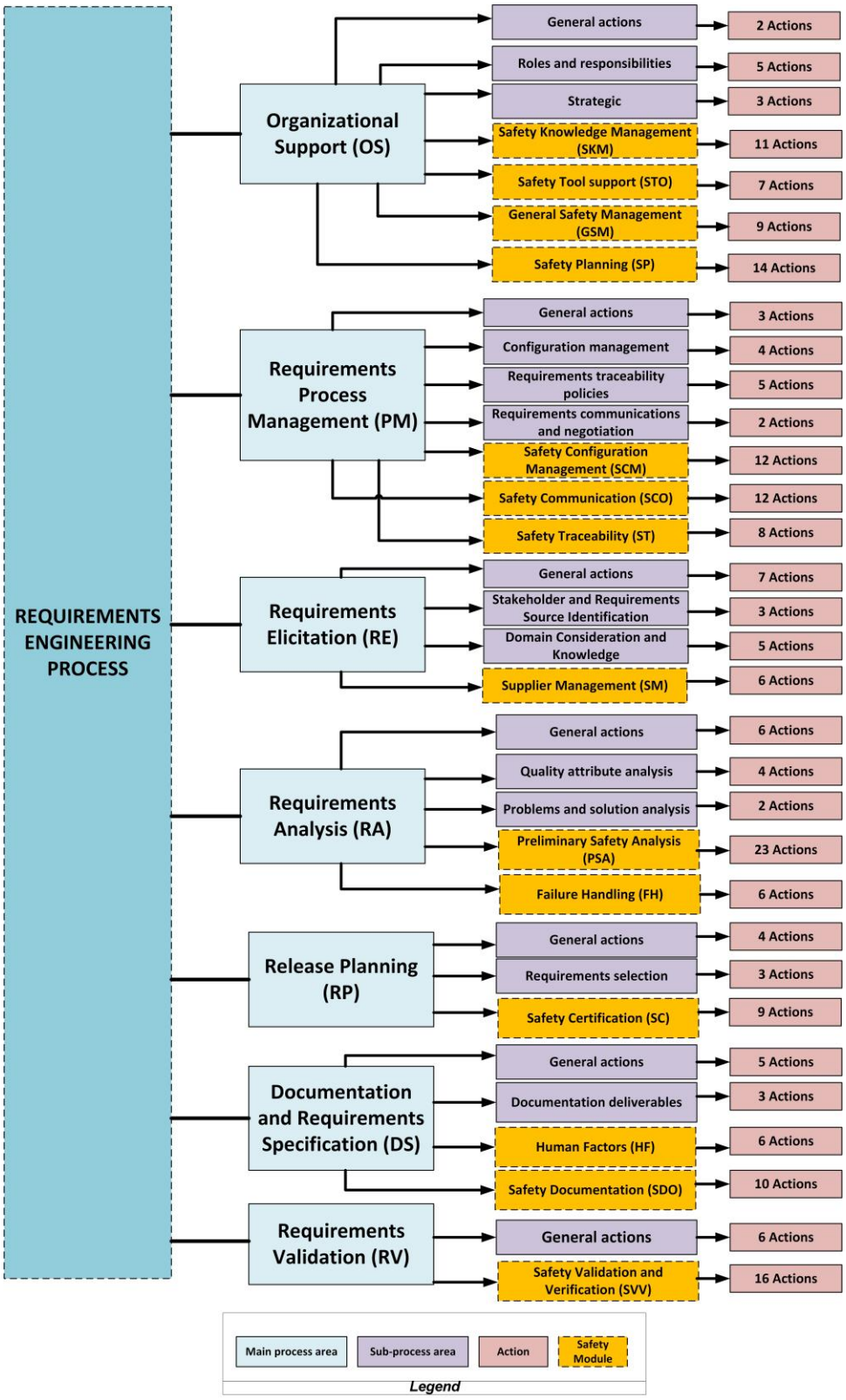

Figure 1. Uni-REPM Safety Module structure and its relationship with Uni-REPM.

2.1. Main Process Area (MPA)

There are seven MPAs in the module, represented here according to the active order in the requirements engineering process:

- **Requirements Elicitation (RE):** it handles actions for discovering and understanding the necessities and desires of costumers in order to communicate them to others stakeholders.
- **Requirements Analysis (RA)**: contains activities to detect errors, create detailed view of requirements as well as to esteem information needed in later activities of RE process.
- **Documentation and Requirements Specification (DS):** addresses how a company structures the requirements and other information collected during elicitation into consistent, accessible and reviewable documents.
- **Requirements Validation (RV):** includes checking the requirements against defined quality standards and the real needs of the several stakeholders. Its aim is to assure that the documented requirements are complete, correct, consistent, and unambiguous.
- **Requirements Process Management (PM)**: contains activities to manage, control requirements change as well as to assure that the process is being followed.
- **Organizational Support (OS)**: assesses the quantity of support provided to RE practices from the surrounding organizations.
- **Release Planning (RP):** comprises important actions to define the optimal set of requirements for a certain release in order to accomplish defined/estimated time and cost goals.

Each MPA has a unique identifier which enables traceability throughout the module. For example, "*Organizational Support*" MPA is referred to as "OS".

2.2. Sub-Process Area (SPA)

Sub-process area (SPA) contains closely related actions, which help to achieve a bigger goal. The unique identifier assigned to each SPA is composed of the MPA identifier to which the SPA attaches and its abbreviation. For example, "OS.SKM" represents a sub-process area called "Safety Knowledge Management (SKM)" which resides under MPA "Organizational Support".

The Safety module is composed by fourteen sub-process area:

- **Safety Planning (SP):** provisions the safety practices and to establish a safety culture in the company.
- **Supplier Management (SM):** is responsible to manage the acquisition of products and services from suppliers external to the project for which shall exist a formal agreement.
- **Preliminary Safety Analysis (PSA):** it addresses the conduction of a preliminary safety analysis to dismiss avoiding wasting effort in next phases of system development.
- **Failure Handling (FH):** it handles issues with failures in system components that can lead to hazardous situations, addition of redundancy as well as protection mechanisms.
- **Safety Validation and Verification (SVV):** it contains actions to requirements validation and the definition of strategies to the verification of requirements aiming to obtain requirements clearly understood and agreed by the relevant stakeholders.

- **Safety Certification (SC):** it has actions related to system certification.
- **General Safety Management (GSM):** it covers project safety management activities related to planning, monitoring, and controlling the project.
- **Safety Configuration Management (SCM):** it addresses the control of content, versions, changes, distribution of safety data, proper management of system artifacts and information important to the organization at several levels of granularity.
- **Safety Communication (SCO):** it aims to improve the safety communication sub process by establishing actions related to many safety terms, methods, process to support the safety analysis and assurance processes.
- **Human Factors (HF):** it handles issues regarding system's users and operators that can lead to hazards and shall be considered during the RE stage of safety-critical system development.
- **Safety Tool support (STO):** is responsible for facilitate the appropriate execution of the corresponding tasks and manage all safety-related information that should be created, recorded and properly visualized.
- **Safety Documentation (SDO):** it has practices to record all information related to system's safety produced in RE phase.
- **Safety Traceability (ST):** it handles the traceability among artifacts helping to determine that the requirements affected by the changes have been completely addressed.
- **Safety Knowledge Management (SKM):** it provides transparency in the development process by making sure that projects and the company have the required knowledge and skills to accomplish project and organizational objectives.

2.3. Action

The smallest unit in the module is called "action" showing a specific good practice. By performing the action, the organization can improve their process and gain certain benefits. For example, an action "*Develop a safety information system to share knowledge in the organization*" once implemented will enable practitioners to share knowledge in the organization improving the communication between them.

Actions also follow the same format to form their unique identifiers. They are identified by the MPA/ SPA under which they reside, followed by an "a" which stands for "action" and their position in the group. For example, "OS.a1" points to the first action which attaches directly to MPA "*Organizational Support*". Another example is "OS.SKM.a1", which means the first action under MPA "*Organizational Support*" and SPA "*Safety Knowledge Management*".

Each action is assigned a certain level depending on its difficulty to implement and essentiality for the requirements engineering process. The level structure will be discussed in detail in section 3.

Example(s) and Supporting Action(s)

Within the description of each Action, there can be **Example(s)** and **Supporting Action(s)**.

The idea of **Example(s)** is to give practitioners suggestions on proven techniques or supporting tools when performing the action. It is worth noticing that the Example item, as the name suggests, is not an exhaustive list. Therefore, companies are not restricted to apply only those in order to fulfill an action.

In addition, the **Supporting Action(s)** provided links to other Actions which will benefit the practitioners when implementing them together. Figure 2 shows a snapshot of the module to illustrate its structure and components.

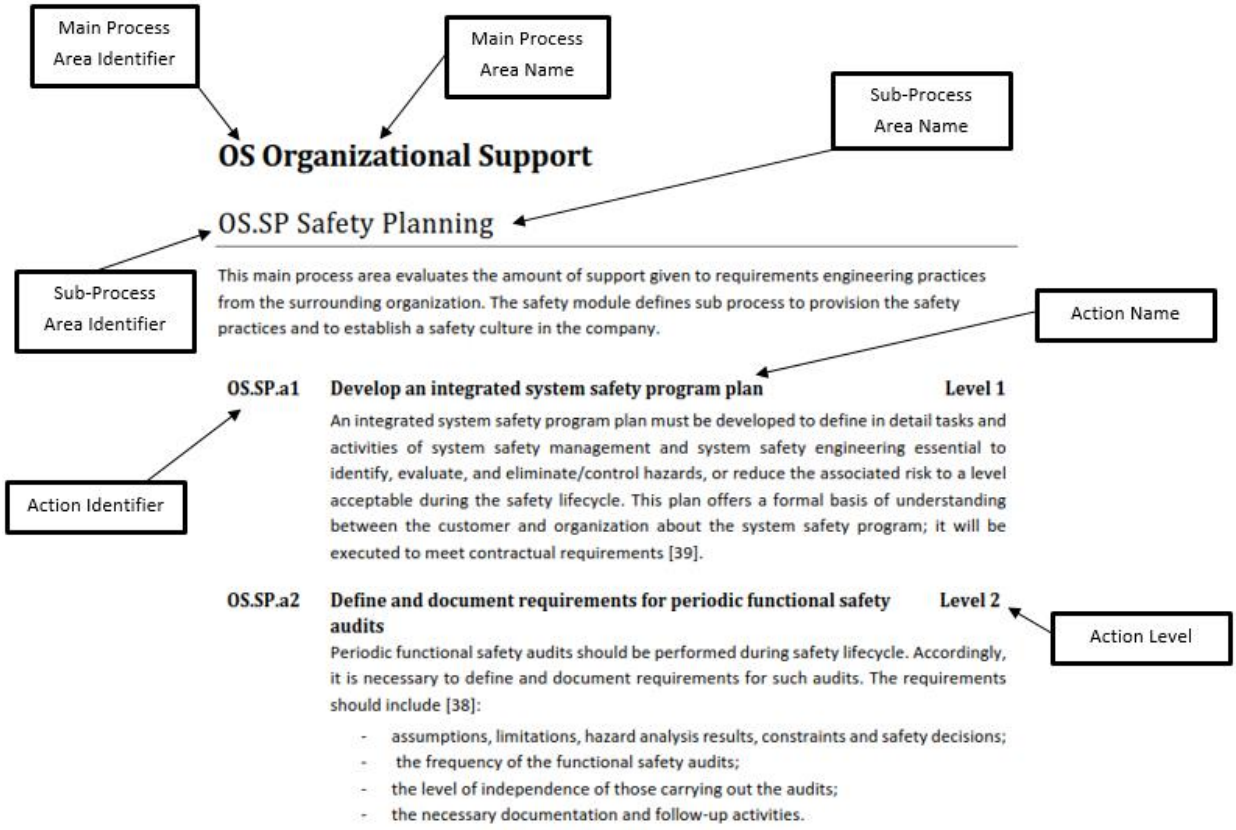

Figure 2. A snapshot of Uni-REPM module.

3. Process maturity

The Safety module follows the ordinal scale to assess the maturity of the process adopted by Uni-REPM. Accordingly, the module has three levels of maturity, namely ***Basic, Intermediate*** and ***Advanced.*** We opted to maintain the likert scale with three levels of Uni-REPM, as adopted by other maturity models [59][61] considering the difficulties users have in choosing among five options with very discrete differences as adopted in many maturity models.

Accordingly, we want users be aware and can clearly distinct among the stages, reducing implications on its application and improving interpretation of stages. This reduced number of maturity levels makes easy practitioners to understand what it means that their RE is assessed to be on a particular maturity level [62].

The levels represent how mature the evaluated process is. It is, however, not applicable to the whole organization maturity since the module scope only resides on the safety concerns in the Requirements Engineering Process. Nevertheless, it is possible to compare two processes in term of maturity using the evaluation results from the module.

The resulting level of a process is constructed from levels of actions performed within such process. As well as in Uni-REPM, in the module, each action is placed under a certain level concerning its essentiality and required skills/cost to carry out. We also considered the dependencies among actions when assigning levels to them, e.g. if action A requires another pre-requisite action to be performed, it must be placed at least at the same or higher level than the pre-requisite action.

Level 1 – Basic

The aim of this level is to achieve a rudimentary repeatable requirements engineering process. The process in this level is defined and followed. Quality of requirements is managed because of relevant stakeholder involvement in elicitation, in-depth requirements analysis and pre-defined document standards.

However, the process does not maintain any kind of communications among stakeholders and within the organization in term of strategies.

Level 2 – Intermediate

In this level, the process is more rigorous because it involves various perspectives and is led by product strategies/goals. Roles and responsibilities for particular tasks are clearly defined and documented. Change requests are handled in the consistent manner throughout the project. Well-informed decisions about requirement selection can be made by analyzing and prioritizing the requirements systematically.

This process still stays in "***present-state***"; meaning that there is no activity performed to collect and analyze data/feedback for future improvement of the process.

Level 3 – Advanced

This level denotes the most mature process. The improvements in the process are shown in the advanced way of capturing requirements, ensuring their high quality, maintaining communications and common understanding among different stakeholders and pro-actively assessing the decision making process.

The process takes into account the "***future-state***" since it not only covers pre-defined and structured procedures but also adequately pay attention on future works (e.g. reusable materials, port-term evaluation, etc.).

4. Module usage

4.1. Who will directly use the module?

Uni-REPM safety module aims to assess the safety maturity in the RE process; hence it can be used by people who are involved in RE process, deeply understand it and be in charge of process improvement in general. Example users can be:

- Requirements Engineer
- Safety Engineer
- System Engineer

- Product Engineer
- Software Engineer
- Quality assurance engineer
- Project manager
- Product manager

4.2. How to use the module?

To assess the maturity of safety in the RE process, the users basically perform a mapping from the actions present in the module to the activities in a real process using the checklist. The checklist is actually a direct transformation of the module into question form. A snapshot of the checklist is shown in Figure 3.

The checklist follows the same structure as the module with questions grouped according to the MPA and SPA. For each action in the module, there is a corresponding question or group of questions to verify if the action is done or not. The Action ID which links the question(s) to the associated action in the module helps the users in case they need to locate the item for further information or clarification.

When answering the questions, the users may encounter one of the following situations:

- The action was deemed vital but was performed partially or not at all in this RE process. It should be marked as **"Incomplete"** (IC)
- The action was completed in this RE process. It should be marked as "**Complete"** (C)
- The action was not necessary or possible to be performed in the process. It should be marked as **"Inapplicable"** (IA)

More about "Inapplicable"

In reality, as organizations and processes vary in their characteristics and environments, they may not benefit from implementing all the actions in the module. Some of the actions are deemed unnecessary to be performed in particular situations of organizations.

For example, in small systems, prototypes may be not useful since the system can be very simple. In this case, the action "*OS.SKM.a4 Evaluate prototypes, requirements and technical UI restrictions (Basic Level)*" might not useful for some companies. If we consider it as "Incomplete", the process may not reach the Basic level because not all actions in this level are fulfilled. This is even more unfair if all other actions in higher maturity levels are completed.

Therefore, companies should not be "punished" if they do not perform a certain nonessential action (in their point of view). In order to take into account this factor, the option "Inapplicable" is devised. In this way, the module fits more real process and the evaluation result is less distorted. Therefore, in some cases, the organization may find some actions only applicable in one of the settings.

Whether an action is "Inapplicable" or not is solely based on the judgment of the project evaluator. Reasons for marking the action with this option should be considered carefully to avoid accidentally skipping an important action. Moreover, lack of time, resource or unawareness cannot be accounted for an "Inapplicable" action.

Action ID	Question	(C)	(IC)	(IA)	Comment / Reason if Inapplicable
OS	**Organizational Support**				
OS.GA	General Actions				
OS.SKM.a1	Do you maintain an infrastructure to share knowledge?				
OS.SKM.a8	Do you reuse the stored knowledge?				

Figure 3. Safety module Checklist snapshot.

4.3. How to read the result?

After answering all questions present in assessment instrument, the user can collect the results for each MPA and consider the following rules.

- For each MPA, all actions at a certain level must be **Completed** (or **Inapplicable**) in order for the MPA to achieve such level.
- For the whole process, all actions at a certain level must be **Completed** (or **Inapplicable**) in order for the process to achieve such level.

An example

The result of MPA “Organizational Support” after evaluating may look like in Table 1.

Table 1. Assessment result in MPA "*Organizational Support*".

Level	Actions in real process		Total actions in OS in Safety Module
	Completed	Inapplicable	
Basic	4	0	2
Intermediate	7	1	6
Advanced	10	2	16

To have a better view, the result can be presented in graph as follows.

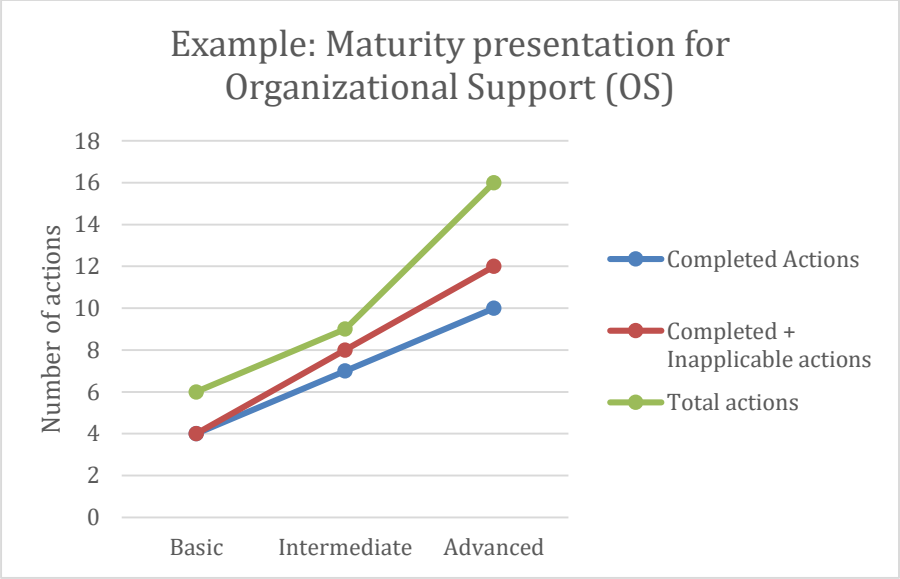

Figure 4. Graphical presentation of assessment results.

The blue line presents actions which were completed. In this case, 4 actions were completed in the Basic Level, 7 actions in Intermediate level and 10 actions in the Advanced level. The red line presents completed actions together with actions that were not performed due to unnecessary or inapplicable reasons.

The distance between the blue line and red line is called the module lag, which represents the number of inapplicable actions. Hence, the module lag shows the applicability of the module in the real setting. In this case, the module lag is fairly small with only two inapplicable actions. This means a high applicability of the module.

Besides, the green line presents the total actions that should be completed in 3 levels of ``Organizational Support" MPA. For example, at Basic level, there are 2 actions that should be finished. The difference between the red line and the green line is important because it denotes the improvement area of the process. It shows how many additional actions should be conducted to achieve a certain level of maturity.

Overall, the graph denotes that, in this MPA, the process has not completed all the actions at Basic level. Hence, according to the above rule, the MPA resides on Level 0. In order to reach the Basic level, two more actions have to be done. If the company aims for Intermediate level, it has to perform two Basic actions and one 2 Intermediate. Similar work can be done with other MPAs to achieve the result for the whole process.

Part III. Safety Module Description

The Safety module extends the UNI-REPM model by adding new safety sub-process. Existing main process areas, sub-process areas, their actions and outcomes were not altered and none were removed. The safety new sub-process areas are presented in Table 2.

Table 2. Overview of new functional safety sub-processes, i.e. extensions, to UNI-REPM.

UNI-REPM MPA	New safety sub-process areas
Requirements Elicitation	Supplier Management
Documentation and Requirements Specification	Human Factors
	Safety Documentation
Requirements Analysis	Preliminary Safety Analysis
	Failure Handling
Release Planning	Safety Certification
Requirements Validation	Safety Validation and Verification
Organizational Support	Safety Planning
	General Safety Management
	Safety Tool support
	Safety Knowledge Management
Requirements Process Management	Safety Configuration Management
	Safety Communication
	Safety Traceability

In the next sections, we provide the description of the module in two views: Sub-Process Area and Maturity Level.

1. Sub-Process Area View

In this section, the module will be presented by sub-process area. The new processes, i.e. extensions, are identified through a postfix, for example ".SM", to the process ID. In order to get a complete process assessment model, each safety sub-process area has safety practices (actions) identified through the main process area ID, sub-process area ID and by adding a postfix ".a#", e.g. RE.SM.a1. Table 3 shows the sub-process and actions and their maturity level of UNI-REPM safety module.

Table 3. Description of UNI-REPM safety module by sub-process area view.

ID	Title	Level
RE	**Requirements Elicitation**	
RE.SM	Supplier Management	
RE.SM.a1	Establish and maintain formal agreements among organization and suppliers	2
RE.SM.a2	Identify and document the products to be acquired	2
RE.SM.a3	Select suppliers and record rationale	2
RE.SM.a4	Specify all external systems and safety-related software	1
RE.SM.a5	Establish and maintain detailed system integration procedures for the external systems and safety-related software	1
RE.SM.a6	Define the safety standards that suppliers must follow	1
DS	**Documentation and Requirements Specification**	
DS.HF	Human Factors	
DS.HF.a1	Construct operator task models	2

ID	Title	Level
DS.HF.a2	Document human factors design and analysis	1
DS.HF.a3	Evaluate prototypes, requirements and technical UI restrictions	1
DS.HF.a4	Model and evaluate operator tasks and component black-box behavior	2
DS.HF.a5	Define interfaces considering ergonomic principles	2
DS.HF.a6	Specify Human Machine Interface requirements	2
DS.SDO	Safety Documentation	
DS.SDO.a1	Record safety decisions and rationale	3
DS.SDO.a2	Ensure that safety requirements are incorporated into system and subsystem specifications, including human-machine interface requirements	1
DS.SDO.a3	Document all lifecycle and modification activities	1
DS.SDO.a4	Develop and document training, operational and software user manuals	2
DS.SDO.a5	Document System Limitations	1
DS.SDO.a6	Provide a safety manual	2
DS.SDO.a7	Document lessons learned	2
DS.SDO.a8	Ensure that safety-related information is incorporated into user and maintenance documents	2
DS.SDO.a9	Maintain hazard and risk analysis results for the system throughout the overall safety lifecycle	3
DS.SDO.a10	Include a summary of safety requirements	1
RA	**Requirements Analysis**	
RA.PSA	Preliminary Safety Analysis	
RA.PSA.a1	Identify and document safety-critical computer software components and units	1
RA.PSA.a2	Simulate the process	3
RA.PSA.a3	Identify and document system hazards	1
RA.PSA.a4	Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)	1
RA.PSA.a5	Specify the type of initiating events that need to be considered	1
RA.PSA.a6	Obtain and document information about the determined hazards (causes, probability, severity, duration, intensity, toxicity, exposure limit, mechanical force, explosive conditions, reactivity, flammability etc.)	1
RA.PSA.a7	Identify and document hazardous materials	1
RA.PSA.a8	Identify and document consequences of hazards, severity categories and affected assets	1
RA.PSA.a9	Conduct risk estimation	1
RA.PSA.a10	Conduct risk evaluation for each identified hazard	1
RA.PSA.a11	Identify and document risk mitigation procedures for each identified hazard	1
RA.PSA.a12	Collect safety requirements from multiple viewpoints	3
RA.PSA.a13	Identify and document pure safety requirements	1
RA.PSA.a14	Identify and document safety-significant requirements and safety integrity levels	1
RA.PSA.a15	Identify and document safety constraints and how they could be violated	1
RA.PSA.a16	Identify and document possible control flaws and inadequate control actions	1
RA.PSA.a17	Identify and document safety functional requirements	1

ID	Title	Level
RA.PSA.a18	Identify and document operational requirements	1
RA.PSA.a19	Perform and document the feasibility evaluation of safety functional requirements	2
RA.PSA.a20	Prioritize hazards and safety requirements	2
RA.PSA.a21	Identify and document analysis and verification requirements, possible safety-interface problems, including the human-machine interface, and operating support requirements	1
RA.PSA.a22	Perform interface analysis, including interfaces within subsystems (such as between safety-critical and non-safety-critical software components)	2
RA.PSA.a23	Consolidate preliminary system safety technical specification	1
RA.FH	Failure Handling	
RA.FH.a1	Define requirements for the avoidance of systematic faults	1
RA.FH.a2	Specify Fault-detection procedures	1
RA.FH.a3	Specify Restart-up procedures	1
RA.FH.a4	Document the system behavioral model	2
RA.FH.a5	Identify and document Common-cause failures (CCF) and how to prevent them	2
RA.FH.a6	Perform reliability and system performance analysis	1
RP	**Release Planning**	
RP.SC	Safety Certification	
RP.SC.a1	Conduct safety audits	2
RP.SC.a2	Demonstrate the preliminary level of safety achieved by the system	1
RP.SC.a3	Evaluate the threat to society from the hazards that cannot be eliminated or avoided	1
RP.SC.a4	Construct preliminary safety and hazard reports	1
RP.SC.a5	Construct preliminary safety cases	1
RP.SC.a6	Demonstrate preliminary compliance with safety standards	2
RP.SC.a7	Ensure that the hazard report is updated with embedded links to the resolution of each hazard, such as safety functional requirements, safety constraints, operational requirements, and system limitations	3
RP.SC.a8	Document the division of responsibility for system certification and compliance with safety standards during safety planning	2
RP.SC.a9	Specify a maintenance plan	1
RV	**Requirements Validation**	
RV.SVV	Safety Validation and Verification	
RV.SVV.a1	Define the safety validation plan for software aspects of system safety	1
RV.SVV.a2	Define the safety verification plan	1
RV.SVV.a3	Define the technical strategy for the validation of external systems and safety-related software	2
RV.SVV.a4	Define pass/fail criteria for accomplishing software validation and verification	2
RV.SVV.a5	Develop safety test plans, test descriptions, test procedures, and validation and verification safety requirements	2
RV.SVV.a6	Define and maintain a software integration test plan	1
RV.SVV.a7	Validate safety-related software aspects	2
RV.SVV.a8	Ensure that there is no potentially hazardous control actions	2
RV.SVV.a9	Perform safety evaluation and verification at the system and subsystem levels	1

ID	Title	Level
RV.SVV.a10	Conduct joint reviews (company and customer)	2
RV.SVV.a11	Ensure that the stakeholders understand software-related system safety requirements and constraints	2
RV.SVV.a12	Document discrepancies between expected and actual results	2
RV.SVV.a13	Verify the behavioral model	2
RV.SVV.a14	Ensure that software requirements and interface specification are consistent	2
RV.SVV.a15	Perform safety inspections	2
RV.SVV.a16	Identify and fix inconsistencies safety requirements specification	2
OS	**Organizational Support**	
OS.SP	Safety Planning	
OS.SP.a1	Develop an integrated system safety program plan	1
OS.SP.a2	Define and document requirements for periodic functional safety audits	2
OS.SP.a3	Define and document the interface between system safety and all other applicable safety disciplines	1
OS.SP.a4	Delineate the scope of safety analysis	1
OS.SP.a5	Establish the hazards auditing and log file	1
OS.SP.a6	Establish working groups and structures	1
OS.SP.a7	Define and document the regulations and safety standards to be followed	1
OS.SP.a8	Identify any certification requirements for software, safety or warning devices or other special safety feature	1
OS.SP.a9	Define and document requirements completeness criteria and safety criteria	3
OS.SP.a10	Review safety experience on similar systems	2
OS.SP.a11	Specify the general safety control structure	3
OS.SP.a12	Specify operating conditions of the machine and installation conditions of the electronic parts	1
OS.SP.a13	Determine the required performance level	1
OS.SP.a14	Identify and document the hazard analysis to be performed; the analytical techniques (qualitative or quantitative) to be used; and depth within the system that each analytical technique will be used (e.g., system level, subsystem level, component level)	1
OS.GSM	General Safety Management	
OS.GSM.a1	Identify and document the system development methodology	1
OS.GSM.a2	Identify and document safety lifecycle for the system development	1
OS.GSM.a3	Identify and document competence requirements for the safety activities	1
OS.GSM.a4	Set safety policy and define safety goals	1
OS.GSM.a5	Identify and document responsibility, accountability and authority	1
OS.GSM.a6	Define system safety program milestones and relate these to major program milestones, program element responsibility, and required inputs and outputs	1
OS.GSM.a7	Use of indicators on engineering documentation to assess the product properties and the development progress	3
OS.GSM.a8	Prepare progress reports in a period of time defined by the project	2
OS.GSM.a9	Monitor project and take corrective actions	2
OS.STO	Safety Tool support	

ID	Title	Level
OS.STO.a1	Use of verification and validation tools	2
OS.STO.a2	Specify justifications for the selection of the off-line support tools	3
OS.STO.a3	Assess offline support tools which can directly or indirectly contribute to the executable code of the safety related system	3
OS.STO.a4	Record information of the tools in the baseline	2
OS.STO.a5	Use of tools with support to cross reference and maintain the traceability among safety information in the software specification	3
OS.STO.a6	Use of computer-aided specification tools	2
OS.STO.a7	Define and use tools to support the safety process and workflow management	3
OS.SKM	Safety Knowledge Management	
OS.SKM.a1	Establish and maintain an infrastructure to share knowledge	3
OS.SKM.a2	Develop a safety information system to share knowledge in the organization	3
OS.SKM.a3	Define control access mechanisms to the safety information system	3
OS.SKM.a4	Maintain employees competence information	3
OS.SKM.a5	Document a strategy to manage the knowledge	2
OS.SKM.a6	Define a lifecycle for projects artifacts	2
OS.SKM.a7	Define and maintain a strategy for reuse	3
OS.SKM.a8	Reuse the stored knowledge	3
OS.SKM.a9	Document the use of stored knowledge	3
OS.SKM.a10	Notify users about problems, new versions and exclusions of artifacts in use	3
OS.SKM.a11	Manage assets	3
PM	**Requirements Process Management**	
PM.SCM	Safety Configuration Management	
PM.SCM.a1	Maintain accurately and with unique identification all safety configuration items and safety information (hazards, safety requirements, risks, etc.)	3
PM.SCM.a2	Define and document change-control procedures	3
PM.SCM.a3	Define and document safety configuration items to be included in the baseline	1
PM.SCM.a4	Document configuration status, release status, the justification (taking account of the impact analysis) for and approval of all modifications, and the details of the modification	3
PM.SCM.a5	Document the release of safety-related software	3
PM.SCM.a6	Perform safety impact analysis on changes	2
PM.SCM.a7	Specify and follow the template for software modification request	1
PM.SCM.a8	Document the procedures for initiating modifications to the safety-related systems, and to obtain approval and authority for modifications	2
PM.SCM.a9	Maintain and make available the software configuration management log	2
PM.SCM.a10	Appoint all deliverable documents according to the rules defined in the Configuration Management Plan	2
PM.SCM.a11	Upload all documents on the safety information system	3
PM.SCO	Safety Communication	
PM.SCO.a1	Establish formal communication channels among different organizational levels	2
PM.SCO.a2	Define a method of exchanging safety information with the suppliers	1

ID	Title	Level
PM.SCO.a3	Establish a common nomenclature	1
PM.SCO.a4	Train people continuously in system engineering and safety techniques (education)	1
PM.SCO.a5	Use of a common safety information system for system specification and safety analysis	3
PM.SCO.a6	Keep stakeholders updated regarding the progress of all safety-related activities	3
PM.SCO.a7	Construct a repository of common hazards	3
PM.SCO.a8	Define and follow templates for system artifacts	1
PM.SCO.a9	Document how conflicts will be resolved	1
PM.SCO.a10	Identify, record and resolve conflicts	1
PM.SCO.a11	Produce all the deliverables documents based on the official document templates	2
PM.SCO.a12	Make available safety-related software specification to every person involved in the lifecycle	1
PM.ST	Safety Traceability	
PM.ST.a1	Define and maintain traceability policies	3
PM.ST.a2	Define and maintain bi-directional traceability between the system safety requirements and the software safety requirements	3
PM.ST.a3	Define and maintain bi-directional traceability between the safety requirements and the perceived safety needs	3
PM.ST.a4	Link and maintain bi-directional traceability between environmental assumptions and the parts of the hazard analysis based on the assumption	3
PM.ST.a5	Define and maintain bi-directional traceability between system and subsystem verification results and system specification	3
PM.ST.a6	Define and maintain bi-directional traceability between validation results and system specification	3
PM.ST.a7	Define and maintain bi-directional traceability among system hazards into components	3
PM.ST.a8	Justify reasons for not traced software requirements	3

RE Requirements Elicitation

Elicitation is the process of discovering, understanding, anticipating and forecasting the needs and wants of the potential stakeholders in order to convey this information to the system developers. The potential stakeholders can include customers, end-users and other people who have the stake in the system development. In the process, the application domain and organizational knowledge are necessary among other things.

RE.SM Supplier Management

The development of safety-critical systems usually requires a combination of internal software and third-party systems. Therefore, in the RE phase, it is necessary to elicit and specify the requirements that suppliers must satisfy.

Suppliers correspond to internal or external organizations that develops, manufactures, or supports products being developed or maintained that will be delivered to other companies or final customers. Suppliers include in-house vendors (i.e., organizations within a company but which are external to the project), fabrication capabilities and laboratories, and commercial vendors [28].

The *Supplier Management* sub-process is responsible to manage the acquisition of products and services from suppliers external to the project for which shall exist a formal agreement. The actions of this sub-process are described below.

RE.SM.a1 **Establish and maintain formal agreements among organization and suppliers** **Level 2**

Formal agreements among organization and suppliers must be established and maintained. A formal agreement is a document legally valid that describe terms and conditions, a list of deliverables, a schedule, budget, and other relevant information.

Supporting action(s)

- RE.SM.a3 Select suppliers and record rationale

RE.SM.a2 **Identify and document the products to be acquired** **Level 2**

The determination of what products or components will be purchased should be based on an analysis of the needs of the project. This analysis begins in the elicitation phase, continues during the design level, ending when the company decides to buy the product.

RE.SM.a3 **Select suppliers and record rationale** **Level 2**

The selection of suppliers and its rationale, for example, advantages and disadvantages, should be recorded. The list of products to be acquired can provide a direction for such selection.

Supporting action(s)

- RE.SM.a2 Identify and document the products to be acquired

RE.SM.a4 Specify all external systems and safety-related software Level 1

The characteristics of all external systems (e.g. data bus, computer, ground interface, communication protocol, the concurrency and real-time model) that interact with the system as well as safety-related software used to implement functions intended to achieve or maintain a safe state in a safety-critical system must be properly documented.

Supporting action(s)

- RE.SM.a2 Identify and document the products to be acquired

RE.SM.a5 Establish and maintain detailed system integration procedures for the external systems and safety-related software Level 1

Detailed system integration procedures, for example the number of iterations to be performed and details of the expected tests and other types of information, for the components of external systems and safety-related software must be established and maintained.

Supporting action(s)

- RE.SM.a4 Specify all external systems and safety-related software

RE.SM.a6 Define the safety standards that suppliers must follow Level 1

The safety standards to be followed by suppliers must be defined and properly specified. This information will be necessary during the construction of safety cases and certification process of the system being developed.

Supporting action(s)

- RE.SM.a1 Establish and maintain formal agreements among organization and suppliers

DS Documentation and Requirements Specification

Documentation and Requirements specification deal with how a company organizes requirements and other knowledge gathered during requirements engineering process into consistent, accessible and reviewable documents. In the safety module, the management of human factors and the documentation of safety issues are the main concern of the sub-process added to this process. The safety requirements specification (SARS) contains the product's detailed functional and safety requirements.

DS.HF Human Factors

Human factors have a significant importance in safety standards since many hazardous situations are caused by system's users and operator due lack of training or unfamiliarity with the operator mental models. Although, the main goals of human-computer interaction are not primarily for safety but to make recommendations and application of technical guidelines [29], the human factors shall be considered during the RE stage of safety-critical system development.

DS.HF.a1 Construct operator task models **Level 2**

Operator's task models impact fundamental dimensions of system usage such as workload, situation awareness, performance, stress, and tiredness, etc. Therefore, such models must be adequately constructed. The representation of such models using visual task-modeling language allows integrated simulation and analysis of the entire system, including human – computer interactions.

DS.HF.a2 Document human factors design and analysis **Level 1**

Developing safety-critical systems requires integrating human factors into the basic RE process, which in turn has important implications for system requirements. The human factors design and analysis should be performed to ensure that the system is designed for the user, regardless the type of user. This analysis should consider the comfort of the users, fit the human body and their cognitive abilities and the system's functionalities. The results of such analysis should be documented.

DS.HF.a3 Evaluate prototypes, requirements and technical UI restrictions **Level 1**

When the first version of system specification is available or whenever occurs changes on it, the prototypes, requirements and technical UI restrictions should be evaluated with the user. This evaluation, which can be with user in labs or using questionnaires, should consider the system specification. If problems in prototypes, in requirements or in user interface restrictions (UI) are identified, new human factors requirements must be specified.

Supporting action(s)

- DS.HF.a1 Construct operator task models
- DS.HF.a2 Document human factors design and analysis

DS.HF.a4 Model and evaluate operator tasks and component black-box behavior **Level 2**

The component black-box behavior describe the inputs and outputs of each component and their relationships only in terms of externally visible behavior. Black-box behavioral specifications as well as operator tasks can be used to maintain the system and to specify and validate changes before the actual development of the system.

Supporting action(s)

- DS.HF.a1 Construct operator task models

DS.HF.a5 Define interfaces considering ergonomic principles **Level 2**

The interfaces of the safety-critical system should consider ergonomic principles to ensure that the system, including the safety-related parts, is easy to use, and so that the operator is not tempted to act in a hazardous manner.

Supporting action(s)

- DS.HF.a6 Specify Human Machine Interface requirements

DS.HF.a6 Specify Human Machine Interface requirements **Level 2**

The Human-Machine Interfaces specify the connection between user and system. Designing a good interface is a challenging RE task since the construction of a well-operable, user-friendly and ergonomic interface presumes great expertise. The human machine interface requirements, including all elements that a user will touch, see, hear, or use to perform safety control functions and receive feedback on those actions, should be described. These requirements allow providing details about the controls by which a user operates the system.

DS.SDO Safety Documentation

Many artifacts are generated during the development of a safety-critical system that are used throughout the development to construct safety cases or documents with certification purposes. Accordingly, all information related to system's safety produced in RE phase must be recorded. This activity can also be done together with members from other phases that will use the information later.

DS.SDO.a1 Record safety decisions and rationale **Level 3**

Safety analysis encompasses trade-offs and decision making to provide safety to the system. Therefore, all safety decisions and rationale for them must be documented and included in the safety requirements specification for later analysis and certification.

Supporting action(s)

- DS.SDO.a9 Maintain hazard and risk analysis results for the system throughout the overall safety lifecycle

DS.SDO.a2 Ensure that safety requirements are incorporated into system and subsystem specifications, including human-machine interface requirements **Level 1**

The safety requirements defined to mitigate the hazards should be traced to (sub) systems and components to improve safety communication and to construct the safety cases.

Supporting action(s)

- DS.HF.a6 Specify Human Machine Interface requirements

DS.SDO.a3 Document all lifecycle and modification activities **Level 1**

The company should define a software and safety lifecycle and record the activities and modification occurred in each of the lifecycle.

Supporting action(s)

- OS.GSM.a2 Identify and document safety lifecycle for the system development

DS.SDO.a4

Develop and document training, operational and software user manuals — Level 2

Training, operational and software user manuals must be developed and properly maintained. These manuals will be updated and improved in the next stages of system development.

DS.SDO.a5

Document System Limitations — Level 1

Sometimes not all hazards and risks are possible or viable to be eliminated or controlled, so, the system is released with limitations (accepted risks). Limitations can be associated, for example, with basic functional requirements, environment assumptions, hazards or hazard causal factors, problems encountered or tradeoffs made during RE. Such limitations should be recorded with links to the pertinent portions of the hazard analysis along with an explanation of why they could not be eliminated or adequately controlled. The limitations are used by management and stakeholders to determine whether the system is adequately safe to use; and, hence, affect both acceptance and system certification.

Supporting action(s)

- DS.SDO.a1 Record safety decisions and rationale
- DS.SDO.a9 Maintain hazard and risk analysis results for the system throughout the overall safety lifecycle

DS.SDO.a6

Provide a safety manual — Level 2

A safety manual describing the functions as well as the inputs and outputs interfaces of an external element must be provided. The manual also should contain the identification of the hardware and/or software configuration of the compliant element to enable configuration management of safety-related system. Moreover, it is also necessary to relate constraints on the use of the element and/or assumptions on which analysis of the behavior or failure rates of the item are based. Such manual may be derived from the supplier's own documentation and records, or may be created or supplemented by the company. If available, reverse engineering can be used.

Supporting action(s)

- RE.SM.a1 Establish and maintain formal agreements among organization and suppliers
- RE.SM.a4 Specify all external systems and safety-related software
- RE.SM.a5 Establish and maintain detailed system integration procedures for the external systems and safety-related software
- RE.SM.a6 Define the safety standards that suppliers must follow

DS.SDO.a7

Document lessons learned **Level 2**

Many times the company develops new versions of existing systems with new functionalities or constructs new systems but in the same area. In this context, a better safety analysis can be conducted by collecting information from previous projects. Hence, the company should document lessons learned to prevent or mitigate risks already identified.

Supporting action(s)

- DS.SDO.a1 Record safety decisions and rationale

DS.SDO.a8

Ensure that safety-related information is incorporated into user and maintenance documents **Level 2**

Safety-related information must be included into user and maintenance documents as long as they are produced. Moreover, periodic reviews should be conducted to ensure that such information were incorporated.

Supporting action(s)

- DS.SDO.a1 Record safety decisions and rationale
- DS.SDO.a9 Maintain hazard and risk analysis results for the system throughout the overall safety lifecycle

DS.SDO.a9

Maintain hazard and risk analysis results for the system throughout the overall safety lifecycle **Level 3**

The results of hazard and risk analysis must be maintained throughout the overall safety lifecycle, from the RE phase to the disposal phase.

DS.SDO.a10

Include a summary of safety requirements **Level 1**

To improve the communication among stakeholders a summary of safety requirements with their associated page numbers in the document must be produced and maintained.

Supporting action(s)

- DS.SDO.a9 Maintain hazard and risk analysis results for the system throughout the overall safety lifecycle

RA Requirements Analysis

Requirements gathered from different sources need to be analyzed to detect incomplete or incorrect ones as well as to estimate necessary information for later activities (e.g. risk, priorities…). It is also necessary to conduct a preliminary safety analysis and failure handling to dismiss avoiding wasting effort in next phases of system development.

RA.PSA Preliminary Safety Analysis

RA.PSA.a1 **Identify and document safety-critical computer software components and units** **Level 1**

Improving system safety requires the identification of safety-critical computer software components and units that demand special attention. Safety engineers and the quality assurance staff will be responsible to monitoring of the strategies to reduce hazardous situations associated with these elements.

RA.PSA.a2 **Simulate the process** **Level 3**

Better safety analysis can be performed by simulating the process related to the system. The process simulation enable modeling complex tasks providing a representative environment to elaborate and test hypotheses. The system can also be simulated by analyzing its inputs and outputs, anticipated occurrences as well as undesired conditions requiring system action.

RA.PSA.a3 **Identify and document system hazards** **Level 1**

The identification of hazards should be identified using appropriate methods and tools for the type of system and be properly recorded.

Possible documents/sources to be consulted or analyzed to achieve this task may be:

- system specification;
- lessons learned;
- pertinent standards and regulations;
- safety design checklists;
- safety related interface considerations among various elements of the system;
- environmental constraints;
- facilities;
- real property installed equipment;
- support equipment and training;
- safety-related equipment
- safeguards; and
- possible malfunctions to the system, subsystems, or software.

Supporting action(s)

- DS.SDO.a7 Document lessons learned
- OS.SP.a14 Identify and document the hazard analysis to be performed; the analytical techniques (qualitative or quantitative) to be used; and depth within the system that each analytical technique will be used (e.g., system level, subsystem level, component level)

RA.PSA.a4 **Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)** **Level 1**

Besides system hazards, a safety-critical system can suffer from hazards, hazardous situations or harmful events due to interaction with other equipment or systems (installed or to be installed). Therefore, it is necessary perform the analysis related to this information.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards

RA.PSA.a5 Specify the type of initiating events that need to be considered Level 1

Hazards generally are initiated by some event. Hence, the type of these event must be considered during safety analysis.

Example of events may be:

- component failures
- procedural faults
- human error; and
- dependent failure mechanisms that can cause hazardous events.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards
- RA.PSA.a4 Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)

RA.PSA.a6 Obtain and document information about the determined hazards (causes, probability, severity, duration, intensity, toxicity, exposure limit, mechanical force, explosive conditions, reactivity, flammability etc.) Level 1

Once hazards are identified, the next step is to specify details about them. Some information are required during the construction of safety cases and certification of the system.

Example of data that should be recorded are [33]:

- cause of hazard
- probability
- severity
- duration
- intensity
- toxicity
- exposure limit
- mechanical force
- explosive conditions
- reactivity

- flammability etc.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards
- RA.PSA.a4 Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)

RA.PSA.a7 Identify and document hazardous materials Level 1

Some safety-critical systems, specially the medical ones, can be constructed using materials that can cause allergic reactions. Therefore, it is necessary to specify any item or substance that, due to its chemical, physical, toxicological, or biological nature, could cause harm to people, equipment, or the environment. Moreover, this information should be present in system specification and available to potential users.

RA.PSA.a8 Identify and document consequences of hazards, severity categories and affected assets Level 1

When a hazardous situation occurs, it may result in consequences for people and environment. Accordingly, the types of such consequences, for example incident and accident, should recorded.

The severity categories may be specified following the classification of safety standards. The MIL-STD-882D [32] for example define four categories:

- Catastrophic
- Critical
- Marginal
- Negligible

Moreover, the affected assets should also be specified.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards
- RA.PSA.a4 Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)
- RA.PSA.a6 Obtain and document information about the determined hazards (causes, probability, severity, duration, intensity, toxicity, exposure limit, mechanical force, explosive conditions, reactivity, flammability etc.)

RA.PSA.a9 Conduct risk estimation Level 1

Hazardous situations can be originated due to failures in system components that are hard to discover by either analysis or test. This difficult can originate the release of systems allowing uncommon hazards.

After the identification of hazards, a risk analysis should be conducted. It involves the risk estimation and risk evaluation. Risk estimation corresponds to the identification of risks presented by hazards, barrier failures and human errors and their quantification.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards
- RA.PSA.a4 Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)
- RA.PSA.a6 Obtain and document information about the determined hazards (causes, probability, severity, duration, intensity, toxicity, exposure limit, mechanical force, explosive conditions, reactivity, flammability etc.)

RA.PSA.a10 Conduct risk evaluation for each identified hazard — Level 1

The risk evaluation addresses decision making about the risk level and its priority during the mitigation specification phase through the application of the criteria developed when the context was established.

The ISO 15998 [33] safety standard recommends the use of risk assessment methodologies such as presented in ISO 14121-1 or IEC 61508-5.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards
- RA.PSA.a4 Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)
- RA.PSA.a6 Obtain and document information about the determined hazards (causes, probability, severity, duration, intensity, toxicity, exposure limit, mechanical force, explosive conditions, reactivity, flammability etc.)

RA.PSA.a11 Identify and document risk mitigation procedures for each identified hazard — Level 1

Risk mitigation procedures should be defined to handle the hazards and reduce the risks previously identified. Examples of procedures are prevention, detection, reaction, and adaptation.

Supporting action(s)

- RA.PSA.a9 Conduct risk estimation
- RA.PSA.a10 Conduct risk evaluation for each identified hazard

RA.PSA.a12 Collect safety requirements from multiple viewpoints — Level 3

The development of safety-critical system requires multidisciplinary teams (computer science, medical, electrical, mechanical, among others) that have different backgrounds and expertise. Accordingly, better safety analysis will be achieved if safety requirements were collected from multiple viewpoints.
The safety requirements can be of different types [34]: pure safety requirements, safety-significant requirements, and safety functional requirements.

Supporting action(s)

- OS.SP.a3 Define and document the interface between system safety and all other applicable safety disciplines
- RA.PSA.a13 Identify and document pure safety requirements
- RA.PSA.a14 Identify and document safety-significant requirements and safety integrity levels
- RA.PSA.a15 Identify and document safety constraints and how they could be violated
- RA.PSA.a16 Identify and document possible control flaws and inadequate control actions
- RA.PSA.a17 Identify and document safety functional requirements
- RA.PSA.a18 Identify and document operational requirements

RA.PSA.a13 Identify and document pure safety requirements Level 1

Pure safety requirements should be identified and specified. These requirements are a kind of quality requirement.

Example

"The system shall not cause more than 3 amount of accidental harm per year."

RA.PSA.a14 Identify and document safety-significant requirements and safety integrity levels Level 1

Sometimes, some requirements are not originally defined to mitigate some hazard, but they can have significant safety ramifications. They are non-safety primary mission requirements and due to their relationship with safety, they should be identified and documented**.**

Safety-significant requirements can be identified based on hazard analysis results and sources of such requirements can be [34]:

- Functional Requirements
- Data Requirements
- Interface Requirements
- Non-safety Quality Requirements

- Constraints

Safety-significant requirements are classified according to the safety integrity level (SIL) which corresponds to a range of safety integrity values representing a category of required safety. In IEC 61508, SIL can be in a range of 1-4 where level 4 has the highest level of safety integrity and level 1 has the lowest.

Example(s)

Requirements for controlling elevator doors.
Requirements to control insulin infusion.

RA.PSA.a15 Identify and document safety constraints and how they could be violated — Level 1

The safety requirements specification may have safety constraints that are engineering decisions that have been chosen to be mandated as a requirement intended to ensure a minimum level of safety. Therefore, any safety-related or relevant constraints between the hardware and the software should be identified and documented.

Example of sources of safety constraints are [34]:

- Architecture constraints
- Design constraints
- Implementation (e.g., coding) constraints
- Testing constraints

Moreover, it is necessary to conduct an analysis about how the safety constraints of a system could be violated and add mechanisms to enforce them.

Supporting action(s)

- RA.PSA.a16 Identify and document possible control flaws and inadequate control actions

RA.PSA.a16 Identify and document possible control flaws and inadequate control actions — Level 1

Following control theory principles, the system must be analyzed to identify possible control flaws and inadequate control actions. Inadequate control actions can be hazardous in four ways [35]:

- A control action required for safety is not provided;
- An unsafe control action is provided;
- A potentially safe control action is provided too late, or out of sequence;
- A correct action is stopped too soon.

RA.PSA.a17 Identify and document safety functional requirements Level 1

Safety functional requirements are functions to be implemented in a safety-critical system that is intended to achieve or maintain a safe state for the system, in respect of a specific hazardous situation. These requirements should be identified and properly specified.

Example

- Emergency core coolant system for nuclear power plant

RA.PSA.a18 Identify and document operational requirements Level 1

Operational requirements, which are the basis for system requirements, of a safety-critical system should be identified and recorded. These requirements describes how to run the system.

Example

- Logging, startup/shutdown controls, monitoring, resource consumption, backup, availability among others.

RA.PSA.a19 Perform and document the feasibility evaluation of safety functional requirements Level 2

Occasionally, the safety functional requirements defined are not viable or impossible to implement. Therefore, stakeholders should conduct a feasibility evaluation of such requirements. In such analysis trade-offs are performed aiming to achieve a best combination of viability, safety and cost. Sometimes, the definition of new safety functional requirements are necessary.

Supporting action(s)

- RA.PSA.a17 Identify and document safety functional requirements
- RA.PSA.a20 Prioritize hazards and safety requirements

RA.PSA.a20 Prioritize hazards and safety requirements Level 2

Hazards in a system have different levels of severity and consequences. The lack of prioritization can severely limit the RE process, and the success of the project, because such activities helps to identify critical requirements and contributes to the decision making process [36]. Therefore, some hazards should have high priority and more resources allocated to mitigate them. In this step, hazards and safety requirements are prioritized and the results recorded.

Supporting action(s)

- RA.PSA.a8 Identify and document consequences of hazards, severity categories and affected assets
- RA.PSA.a17 Identify and document safety functional requirements

RA.PSA.a21 Identify and document analysis and verification requirements, possible safety-interface problems, including the human-machine interface, and operating support requirements **Level 1**

In this step, analysis and verification requirements, possible safety-interface problems, including the human-machine interface, and operating support requirements should be defined. The specification of such requirements in the RE process is necessary to avoid defining a hazard that may be implemented correctly but whose test is impossible or very costly [37].

RA.PSA.a22 Perform interface analysis, including interfaces within subsystems (such as between safety-critical and non-safety-critical software components) **Level 2**

In this step, the hazard analysis should be reviewed and updated to consider problems with hardware-software and their interfaces.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards
- RA.PSA.a4 Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)

RA.PSA.a23 Consolidate preliminary system safety technical specification **Level 1**

In this step, it is necessary to ensure that the results of all analysis conducted and the information identified are consolidated in a preliminary system safety technical specification.

Hence, it is important to specify and manage these faults. The safety module has a sub process to handle such failures.

RA.FH Failure Handling

RA.FH.a1 Define requirements for the avoidance of systematic faults **Level 1**

Systematic faults can happen in the system due to their complexity. In this step, an analysis should be conducted to define requirements for the avoidance or control of those faults. The definition of such requirements depend on the expertise of the requirements engineer and judgment from practical experience gained in industry.

Supporting action(s)

- RA.FH.a2 Specify Fault-detection procedures

- RA.FH.a3 Specify Restart-up procedures

RA.FH.a2 Specify Fault-detection procedures Level 1

To avoid hazards and maintain a safe state in the system, it is important to monitor a system, identifying when a fault has occurred, and presenting its type and location. This early detection of a fault contributes to avoid systematic faults and providing time to the system to recover from the fault.

RA.FH.a3 Specify Restart-up procedures Level 1

Sometimes, hazards can be eliminated by taking restart-up procedures. This step of the safety module concerns to the specification of such automatic procedures.

RA.FH.a4 Document the system behavioral model Level 2

The specification of the system behavioral model allows to verifying early its behavior against the one expected. This analysis contributes to detect early the errors and inconsistencies in the system specification as well as to anticipate the correct behavior of the system.

RA.FH.a5 Identify and document Common-cause failures (CCF) and how to prevent them Level 2

Some failures may have a shared cause and its repeatability is known. Such failures are called Common-cause failures (CCF) and due to the presence of many electronic parts in the system, they should be identified and documented.

RA.FH.a6 Perform reliability and system performance analysis Level 1

The time to failure as well as to repair some component impact in system recovery and avoidance of hazardous situations. Accordingly, reliability and system performance analysis should be conducted and its results recorded.

Supporting action(s)

- OS.SP.a13 Determine the required performance level

RP Release planning

Release planning consists in determining the optimal set of requirements for a certain release to be implemented at a defined/estimated time and cost to achieve some goals. A careful release planning is necessary to avoid risky situations, fail to achieve planned goals or miss the time-to-market. Besides the sub processes and actions already present in UNI-REPM, the module defines a new one related to system certification.

RP.SC Safety Certification

RP.SC.a1 Conduct safety audits **Level 2**

Safety audits should be conducted to examine whether the requirements are being achieved and the desired level of safety is preserved. This step should be a periodic activity during the RE process as well as the next stages of system development.

Supporting action(s)

- OS.SP.a2 Define and document requirements for periodic functional safety audits

RP.SC.a2 Demonstrate the preliminary level of safety achieved by the system **Level 1**

From the results of safety audits is possible to demonstrate the preliminary level of safety achieved by the system. The level should be compared against the one desired and can be improved still in RE process or in the next stages of development.

Supporting action(s)

- RA.SC.a1 Conduct safety audits
- OS.SP.a7 Define and document the regulations and safety standards to be followed

RP.SC.a3 Evaluate the threat to society from the hazards that cannot be eliminated or avoided **Level 1**

Stakeholders should be aware of the risks caused by hazards that cannot be eliminated or avoided and are present in the system. Hence, the threats to society should be evaluated and properly documented.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards
- RA.PSA.a4 Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)
- RA.PSA.a6 Obtain and document information about the determined hazards (causes, probability, severity, duration, intensity, toxicity, exposure limit, mechanical force, explosive conditions, reactivity, flammability etc.)
- RA.PSA.a9 Conduct risk estimation
- RA.PSA.a10 Conduct risk evaluation for each identified hazard

RP.SC.a4 Construct preliminary safety and hazard reports **Level 1**

During the development of safety-critical systems results in many iterations of hazard analysis, that generates a lot of safety and hazard reports. In RE phase, a preliminary version of such documents should be constructed and updated during system lifecycle.

RP.SC.a5 Construct preliminary safety cases Level 1

At the end of RE stage, all information gathered during safety and hazard analysis should be used to construct preliminary safety cases.

Supporting action(s)

- RA.PSA.a3 Identify and document system hazards

RP.SC.a6 Demonstrate preliminary compliance with safety standards Level 2

The safety level achieved at RE phase should be used to demonstrate preliminary compliance with safety standards. The demonstration may be performed by developing a document describing the safety requirements, listing the safety standards and system specifications containing requirements to be satisfy by suppliers among other relevant information.

Supporting action(s)

- RA.SC.a2 Demonstrate the preliminary level of safety achieved by the system
- DS.SDO.a10 Include a summary of safety requirements
- RE.SM.a6 Define the safety standards that suppliers must follow
- OS.SP.a8 Identify any certification requirements for software, safety or warning devices or other special safety feature

RP.SC.a7 Ensure that the hazard report is updated with embedded links to the resolution of each hazard, such as safety functional requirements, safety constraints, operational requirements, and system limitations Level 3

The information about hazards should be easy to find to improve the communication among stakeholders and the traceability in the development process. Accordingly, safety functional requirements, safety constraints, operational requirements, and system limitations should be inserted in the hazard report and periodically updated.

Supporting action(s)

- RA.SC.a4 Construct preliminary safety and hazard reports
- PM.SCM.a1 Maintain accurately and with unique identification all safety configuration items and safety information (hazards, safety requirements, risks, etc.)
- OS.STO.a5 Use of tools with support to cross reference and maintain the traceability among safety information in the software specification

RP.SC.a8 Document the division of responsibility for system certification and compliance with safety standards during safety planning Level 2

Division of responsibility is necessary in the development of safety-critical systems especially in large and complex projects. This division of activities among personnel should

be documented during safety planning and include the specification of people responsible for system certification and to demonstrate compliance with safety standards.

Supporting action(s)

- OS.SP.a7 Define and document the regulations and safety standards to be followed

RP.SC.a9 Specify a maintenance plan **Level 1**

A maintenance plan is necessary to release of a safety-critical system. This plan should describe the development and testing activities required to be undertaken on each new release of software including the obsolescence of development equipment, test environments and software among other relevant information.

RV Requirements Validation

Requirements validation includes the inspection of the produced documents against defined safety and quality standards and the needs of stakeholders. In the safety module, a sub process to plan the verification and validation activities was added since they often run concurrently and may use portions of the same environment.

RV.SVV Safety Validation and Verification

In the Safety Validation and Verification (V&V) there are actions to validation of the requirements and the definition of strategies to the verification of requirements. V&V activities should be available early in the development process so that the safety requirements are clearly understood and agreed by the relevant stakeholders.

RV.SVV.a1 Define the safety validation plan for software aspects of system safety **Level 1**

The objective of this action is to define a safety validation plan for software aspects of system safety. This plan should contain [38]:

- details of when the validation will be conducted;
- details of personnel responsible for performing the validation;
- identification of the relevant modes of system operation such as preparation for use including setting and adjustment, startup, automatic, manual, re-setting, shut down, maintenance, and uncommon conditions;
- identification of the safety-significant software which needs to be validated;
- the technical strategy for the validation;
- the required environment in which the validation activities will be performed;
- the pass/fail criteria;

- the policies and procedures for evaluating the results of the validation, particularly failures.

Supporting action(s)

- RA.SVV.a3 Define the technical strategy for the validation of external systems and safety-related software
- RA.SVV.a4 Define pass/fail criteria for accomplishing software validation and verification

RV.SVV.a2 Define the safety verification plan Level 1

The demonstration that safety will be properly achieved encompasses the definition of a safety verification plan. This plan comprises planning inspection, testing, analyses, and demonstration activities and should describe the following information [28][39][40]:

- methods of verification (for example, inspections, peer reviews, audits, walkthroughs, analyses, simulations, testing, and demonstrations);
- support tools, test equipment and software, simulations, prototypes, and facilities;
- safety test specifications;
- required outcome of the tests for compliance;
- chronology of the tests.

Supporting action(s)

- RA.SVV.a5 Develop safety test plans, test descriptions, test procedures, and validation and verification safety requirements.
- RA.PSA.a21 Identify and document analysis and verification requirements, possible safety-interface problems, including the human-machine interface, and operating support requirements

RV.SVV.a3 Define the technical strategy for the validation of external systems and safety-related software Level 2

A technical strategy for the validation (for example analytical methods, statistical tests etc.) should be defined and the rationale for choosing it recorded. The strategy should include [38]:

- choice of manual or automated techniques or both;
- choice of static or dynamic techniques or both;
- choice of analytical or statistical techniques or both;
- choice of acceptance criteria based on objective factors or expert judgment or both.

RV.SVV.a4 Define pass/fail criteria for accomplishing software validation and verification Level 2

A part of safety V&V activities consists in defining pass/fail criteria for accomplishing them. The criteria should address [38]:

- the required input signals with their sequences and their values;
- the anticipated output signals with their sequences and their values;
- other acceptance criteria, for example memory usage, timing and value tolerances.

RV.SVV.a5 Develop safety test plans, test descriptions, test procedures, and validation and verification safety requirements **Level 2**

The goal of this step is to define and document preliminary versions of safety test plans, test descriptions, test procedures, and validation and verification of safety requirements. The definition of such documents and requirements to be used in V&V activities aims to ensure that no hazards are introduced by test procedures [37]. Therefore, this should be careful planned and begin early in the development process.

Supporting action(s)

- RA.PSA.a21 Identify and document analysis and verification requirements, possible safety-interface problems, including the human-machine interface, and operating support requirements

RV.SVV.a6 Define and maintain a software integration test plan **Level 1**

Since there are many systems and subsystems as well as third-party software and equipment communicating with the safety-critical system it is necessary to define and maintain a software integration test plan. A successful integration strategy should use a combination of techniques, depending on the complexity of components [28].

Some factors to be considered during the elaboration of this plan are availability of the product components, test equipment, procedures, integration environment, and personnel skills [28].

RV.SVV.a7 Validate safety-related software aspects **Level 2**

The safety-related software aspects described in the safety validation plan should be validated and the results documented.

Supporting action(s)

- RA.SVV.a1 Define the safety validation plan for software aspects of system safety

RV.SVV.a8 Ensure that there is no potentially hazardous control actions **Level 2**

The aim of this step is to analyze whether the safety control actions provided in the system design previously defined there is no potential for inadequate control, leading to a hazard.

Supporting action(s)

- RA.PSA.a16 Identify and document possible control flaws and inadequate control actions

RV.SVV.a9 Perform safety evaluation and verification at the system and subsystem levels **Level 1**

The safety evaluation and verification of the safety-critical system should be performed at system and subsystem levels to ensure that there is no hazardous situation remains in the system.

Supporting action(s)

- RA.PSA.a21 Identify and document analysis and verification requirements, possible safety-interface problems, including the human-machine interface, and operating support requirements

RV.SVV.a10 Conduct joint reviews (company and customer) **Level 2**

The validation and verification of the system should be performed in meeting with company and customer together. Conducting non-jointly reviews rises the risk to find late disagreements among stakeholders on the product capability or quality, causing substantial reengineering and increasing its cost and time to develop [41].

RV.SVV.a11 Ensure that the stakeholders understand software-related system safety requirements and constraints **Level 2**

Stakeholders involved in the development of a safety-critical system, particularly RE engineers, should understand the software-related system safety requirements and constraints in order to produce better system specification. These requirements should not be merely included in the specification, it is necessary to properly and clearly specify them in details. This will contribute to avoid that developers or other stakeholders involuntarily disable or override system safety features or implement the functionalities erroneously [30].

RV.SVV.a12 Document discrepancies between expected and actual results **Level 2**

Any discrepancies between expected and obtained results of V&V should be documented. It is also necessary to record the analysis made of such discrepancies such as the decisions taken about continuing the validation, the change requests and the return to an earlier part of system development [38].

Supporting action(s)

- RA.SVV.a7 Validate safety-related software aspects
-

RV.SVV.a13 Verify the behavioral model **Level 2**

The verification of system behavior should use the system behavioral model defined previously aiming to ensure the correctness of the system or detect errors and inconsistencies in the system specification.

Supporting action(s)

- RA.FH.a4 Document the system behavioral model

RV.SVV.a14 Ensure that software requirements and interface specification are consistent **Level 2**

The objective of this action is to analyze whether the software requirements and interface specification are compatible and they do not have contradictory issues. The non-consistent parts should be documented and corrected.

RV.SVV.a15 Perform safety inspections **Level 2**

Stakeholders should implement controls and to inspect the RE process and operations in order to discover and correct any additional hazards [30].

RV.SVV.a16 Identify and fix inconsistencies safety requirements specification **Level 2**

The safety requirements specification should be examined in order to find inconsistencies that must be recorded and solved. The documentation of such inconsistencies should include the sources, conditions, rationales, as well as corrective action requirements and actions.

Supporting action(s)

- RV.SVV.a12 Document discrepancies between expected and actual results
- RV.SVV.a13 Verify the behavioral model

OS Organizational Support

OS.SP Safety Planning

This main process area evaluates the amount of support given to requirements engineering practices from the surrounding organization. The safety module defines sub process to provision the safety practices and to establish a safety culture in the company.

OS.SP.a1 Develop an integrated system safety program plan **Level 1**

An integrated system safety program plan must be developed to define in detail tasks and activities of system safety management and system safety engineering essential to identify, evaluate, and eliminate/control hazards, or reduce the associated risk to a level acceptable during the safety lifecycle. This plan offers a formal basis of understanding

between the customer and organization about the system safety program; it will be executed to meet contractual requirements [39].

OS.SP.a2 Define and document requirements for periodic functional safety audits **Level 2**

Periodic functional safety audits should be performed during safety lifecycle. Accordingly, it is necessary to define and document requirements for such audits. The requirements should include [38]:

- assumptions, limitations, hazard analysis results, constraints and safety decisions;
- the frequency of the functional safety audits;
- the level of independence of those carrying out the audits;
- the necessary documentation and follow-up activities.

OS.SP.a3 Define and document the interface between system safety and all other applicable safety disciplines **Level 1**

Considering that there are many disciplines involved in the development of a safety-critical system, the interface between system safety and other safety disciplines such as nuclear, range, explosive, chemical, biological, among others should be defined and recorded.

OS.SP.a4 Delineate the scope of safety analysis **Level 1**

At the very beginning of RE process, the scope and objectives of safety analysis should be defined. This includes an analysis of system boundaries, assumptions to be considered as well as data/information sources and documents to be consulted.

OS.SP.a5 Establish the hazards auditing and log file **Level 1**

The template for the hazards auditing and log file should be created. This file will be periodically updated and should contain corrective actions, waivers, and verification efforts [30].

OS.SP.a6 Establish working groups and structures **Level 1**

In complex systems, special organizational structures such as the definition of working groups that are necessary but do not already exist must be established at this step.

OS.SP.a7 Define and document the regulations and safety standards to be followed **Level 1**

The regulations and safety standards to be followed should be defined and documented. Compliance with such standards is necessary for the certification and release of many safety-critical systems.

OS.SP.a8 Identify any certification requirements for software, safety or warning devices or other special safety feature **Level 1**

The certification requirements for software, safety or warning devices or other special safety features should be identified and documented in this step.

Safety features or devices are define to protect the system when it is not possible to eliminate the hazard. Warning devices, on the other hand, are used to alert personnel to the particular hazard if safety devices do not adequately lower the risk of the hazard. These certification requirements will be used to demonstrate the level of safety achieved by the system and compliance with safety standards.

OS.SP.a9 Define and document requirements completeness criteria and safety criteria — Level 3

Ensuring completeness in a system is a challenging task. A system must not be complete in the mathematical sense, but rather in the sense of a lack of ambiguity. Accordingly, the system specification may be sufficiently complete with respect to safety without being absolutely complete: it just have to achieve the safe behavior in all circumstances in which the system operates [30]. In this step, criteria for requirements completeness and safety should be defined.

OS.SP.a10 Review safety experience on similar systems — Level 2

Lessons learned and safety experience on similar systems of the stakeholders should be reviewed, including mishap/incident hazard tracking logs (if accessible), among other information to identify possible sources of hazards and their risks.

Supporting action(s)

- DS.SDO.a7 Document lessons learned

OS.SP.a11 Specify the general safety control structure — Level 3

Safety-critical systems can be described as hierarchical structures, where each level imposes constraints on the activity of the level beneath it [30]. Such structures describe control processes that should enforce the safety constraints for which the control process is responsible. The determination of a safety control structure is important for safety analysis since accidents occur when these processes provide inadequate control and the safety constraints are violated in the behavior of the lower-level components.

For details about how to elaborate the safety control structure, please see [30].

OS.SP.a12 Specify operating conditions of the machine and installation conditions of the electronic parts — Level 1

Some operating conditions of the machine and installation conditions of the electronic parts as well as other environmental conditions should be specified by the company. This specification may include:

- Environment temperature and humidity
- Degree of protection
- Electromagnetic compatibility

- Mechanical vibration and shock
- Emergency stop function

OS.SP.a13 Determine the required performance level **Level 1**

The performance level that should be satisfied by the system in order to achieve the required risk reduction for each safety requirements should be determined and recorded. This performance level will be used in the reliability analysis of the system.

OS.SP.a14 Identify and document the hazard analysis to be performed; the analytical techniques (qualitative or quantitative) to be used; and depth within the system that each analytical technique will be used (e.g., system level, subsystem level, component level) **Level 1**

The techniques to be used in hazard analysis should be identified. The techniques are classified as qualitative or quantitative. Qualitative analysis concerns with examining the causal relations between events and states in sequences connecting failures of components to hazard states of the system [43]. In the quantitative safety analysis, probabilities (or probability density functions) are assigned to the events in the chain and an overall likelihood of a loss is calculated [30].

The choice of such techniques depend on [31][38] their goals and limitations (i.e., the level of uncertainty, possible unexpected outcomes, assumptions, team knowledge, system complexity, the application sector and its accepted good practices, legal and safety regulatory requirements; and the availability of accurate data upon which the hazard and risk analysis is to be based.

Moreover, the depth within the system that each analytical technique will be used should be specified. The level can be associated for example with [39]: the system, subsystem, components, software, hazardous materials, personnel, ground support equipment, non-developmental items, facilities, and their interrelationship in the logistic support, training, maintenance, operational environments.

OS.GSM General Safety Management

The general safety management sub process covers the project safety management activities related to planning, monitoring, and controlling the project.

OS.GSM.a1 Identify and document the system development methodology **Level 1**

The system development methodology should be defined and properly documented. There are different types of process models to develop software such as traditional methodologies (waterfall model), agile methodologies (XP, Scrum, FDD e Crystal), evolutionary (incremental, prototyping, spiral), and emergent methodologies (based on

reuse, components) among others. The company should choose the one that most fit the project goals and needs of organization.

OS.GSM.a2 Identify and document safety lifecycle for the system development Level 1

A safety lifecycle should be defined by the company and followed during system development.

Example

- Initial concept, design, implementation, operation and maintenance, and disposal [38].

OS.GSM.a3 Identify and document competence requirements for the safety activities Level 1

The competence requirements for the safety activities during the project should be determined. These requirements depends on the knowledge and skills of the employees available to support the development of the project [38]. A two-dimensional matrix with the competences along one-axis and project activities along the other axis may be a suitable format for achieving this identification [38].

Some factors impacts the definition of the competence requirements [38]:

- responsibilities
- level of supervision required
- potential consequences in the event of failure of systems
- novelty of the design
- previous experience and its relevance to the specific duties to be performed and the technology being employed
- type of competence appropriate to the circumstances
- safety engineering knowledge appropriate to the technology
- knowledge of the legal and safety standards
- relevance of qualifications to specific activities to be performed.

Supporting action(s)

- OS.SKM.a4 Maintain employees competence information

OS.GSM.a4 Set safety policy and define safety goals Level 1

Safety Policy, which correspond to strategic decision that establishes a safety goal [34], should be defined. The description of such information may include the relationships of safety to other organizational goals and provide the scope for the discretion, initiative, and judgment in deciding what should be done in specific situations [37].

OS.GSM.a5 Identify and document responsibility, accountability and authority Level 1

Responsibility, accountability and authority for which activity to be performed during development should be assigned and documented.

Supporting action(s)

- OS.SKM.a4 Maintain employees competence information

OS.GSM.a6 Define system safety program milestones and relate these to major program milestones, program element responsibility, and required inputs and outputs **Level 1**

A schedule of system safety activities including required inputs and outputs, start and completion dates that support the RE process should be determined. This schedule will contain the system safety program milestones and the relationships to major program milestones, program element responsibility.

Supporting action(s)

- OS.GSM.a5 Identify and document responsibility, accountability and authority

OS.GSM.a7 Use of indicators on engineering documentation to assess the product properties and the development progress **Level 3**

Indicators about the percentage of requirements allocation, implement, verification, and about the engineering documentation to assess the product properties and the development progress should be identified and recorded.

OS.GSM.a8 Prepare progress reports in a period of time defined by the project **Level 2**

Progress reports are the basis for monitoring activities, communicating status, and taking corrective action. Progress is defined by comparing actual work product and task attributes, effort, cost, and schedule to the plan at prescribed milestones or control levels within the project schedule or work breakdown structure [28]. The elaboration of these reports in a period of time defined by the project allows taking corrective actions early.

The progress reports may describe the implementation status of recommended mitigation measures [44], hazard status among other information.

Supporting action(s)

- OS.GSM.a7 Use of indicators on engineering documentation to assess the product properties and the development progress

OS.GSM.a9 Monitor project and take corrective actions **Level 2**

The defined indicators and the progress reports should be used to monitor the project and take corrective actions when progress varies significantly from that planned. Corrective action may include [28]:

- changing the process(es), changing the plan, or both;
- adjusting resources, including people, tools, and other resources;
- negotiating changes to the established commitments;
- changing the requirements and standards that have to be satisfied;

- finishing the project if necessary.

Supporting action(s)

- OS.GSM.a8 Progress reports should be prepared in a period of time defined by the project

OS.STO Safety Tool support

The RE process is better conducted when supported by adequate tools. In order to be able to facilitate the appropriate execution of the corresponding tasks and manage all safety-related information that should be created, recorded and properly visualized, the module has a sub process to handle these issues.

OS.STO.a1 Use of verification and validation tools — Level 2

Tools to be used during the verification and validation such as static code analyzers, test coverage monitors, theorem proving assistants, and simulators should be determined and their use documented.

OS.STO.a2 Specify justifications for the selection of the off-line support tools — Level 3

The reasons for choosing off-line support tools must be recorded. These tools can be of three types [38]:

1. the ones that generates no outputs which can directly or indirectly contribute to the executable code (including data) of the safety related system, for example, text editors or a requirements or design support tool with no automatic code generation capabilities; configuration control tools;
2. tools that supports the test or verification of the design or executable code, where errors in the tool can fail to reveal defects but cannot directly create errors in the executable software such as test harness generators, test coverage measurement tools; and static analysis tools;
3. the ones that generate outputs which can directly or indirectly contribute to the executable code of the safety related system. Examples of these types may be an optimizing compiler or a compiler that incorporates an executable run-time package into the executable code.

OS.STO.a3 Assess offline support tools which can directly or indirectly contribute to the executable code of the safety related system — Level 3

The off-line support tools selected previously should be evaluated to determine the level of reliance that can be provided by the tools, and their potential failure mechanisms that may affect the executable software. In case of identifying such mechanisms, they must be documented and suitable mitigation procedures must be carried out.

Supporting action(s)

- OS.STO.a2 Specify justifications for the selection of the off-line support tools

OS.STO.a4 Record information of the tools in the baseline **Level 2**

Information about the tools (such as version, installation and execution requirements, name of vendor) used in each baseline must be recorded.

OS.STO.a5 Use of tools with support to cross reference and maintain the traceability among safety information in the software specification **Level 3**

Cross referencing is fundamental for establish and maintain traceability among safety information in the software specification. Therefore, it is necessary to select and use tools that supports this feature.

Supporting action(s)

- OS.STO.a6 Use of computer-aided specification tools

OS.STO.a6 Use of computer-aided specification tools **Level 2**

Use of computer-aided tools contributes for developing high-quality systems since methods for the development of systems together with automated mechanisms can be provided. Such tools facilitates the development, reduce the probability of introducing errors in the system through the use of syntax checks, and other functionalities.

OS.STO.a7 Define and use tools to support the safety process and workflow management **Level 3**

Project management activities can be facilitated using tools to support the safety process and workflow management. Accordingly, the tools that will be used by the project should be defined and documented.

OS.SKM Safety Knowledge Management

The Safety Knowledge Management sub process area provides transparency in the development process by make sure that projects and the company have the required knowledge and skills to accomplish project and organizational objectives. The goal is to guarantee the effective application of project resources (people, knowledge and skill) against the organization's needs.

OS.SKM.a1 Establish and maintain an infrastructure to share knowledge **Level 3**

Collecting and disseminating knowledge about safety concerns across organizational levels can improve safety practices [31]. To achieve this, it is necessary to establish and maintain an infrastructure to support the system capable of sharing knowledge.

OS.SKM.a2 Develop a safety information system to share knowledge in the organization **Level 3**

A safety information system capable of maintain the organization knowledge into a single database contributes to better integration of documents, and teams. Among the benefits a safety information system are a more efficient analysis of tasks and hazards,

better transfer of data with subsequent methods of risk quantification, and better monitoring of safety measures [31].

Supporting action(s)

- OS.SKM.a1 Establish and maintain an infrastructure to share knowledge

OS.SKM.a3 Define control access mechanisms to the safety information system — Level 3

Control access mechanisms to the safety information system should be implemented to enable stakeholders locate and consume only the data adequate for their roles.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization

OS.SKM.a4 Maintain employees competence information — Level 3

The competence, i.e. skills, previous training, technical knowledge, experience and qualifications of company employees should be maintained in the safety information system. This information will be used to identify and document competence requirements for the safety activities, allocate people in teams and responsibility.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization

OS.SKM.a5 Document a strategy to manage the knowledge — Level 2

The strategy to manage the knowledge such as procedures to insert information in the system, personnel responsible for such activity, periodicity of updates must be defined and document.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization

OS.SKM.a6 Define a lifecycle for projects artifacts — Level 2

A lifecycle of project artifacts describing the possible states in which an artifact can be located should be defined and documented.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization

OS.SKM.a7 Define and maintain a strategy for reuse — Level 3

The data stored in the safety information system should be reused to reduce time of development, costs and develop better systems. A strategy for reuse should be defined describing in details the procedures for conducting such activity.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization

OS.SKM.a8 Reuse the stored knowledge — Level 3

The reuse strategy defined must be followed and the stored knowledge should be reused.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization
- OS.SKM.a7 Define and maintain a strategy for reuse

OS.SKM.a9 Document the use of stored knowledge — Level 3

The use of artifacts in a given moment should be documented to improve the communication among stakeholders. The registration that an artifact is being used allows notifying users about problems, new versions and exclusions of artifacts in use.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization

OS.SKM.a10 Notify users about problems, new versions and exclusions of artifacts in use — Level 3

The safety information system should notify the users about problems, updates and exclusions that many occur with artifacts in use.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization
- OS.SKM.a9 Document the use of stored knowledge

OS.SKM.a11 Manage assets — Level 3

The assets of the organization and the system, for example people, property, environment or service should be documented and managed.

Supporting action(s)

- OS.SKM.a2 Develop a safety information system to share knowledge in the organization

PM Requirements Process Management

The requirements process management covers all the activities to manage and control requirements change as well as to ensure the creation, control, and evolution of the processes, as well as coherence among team members. The safety module added three new areas: Safety Configuration Management, Safety Communication, and Safety Traceability.

PM. SCM Safety Configuration Management

The safety configuration management addresses the control of content, versions, changes, distribution of safety data, proper management of system artifacts and information important to the organization at several levels of granularity. Examples of artifacts that may be placed under configuration management include plans, process descriptions, safety requirements, models, system specification, system data files, and system technical publications among other information [28].

PM.SCM.a1 **Maintain accurately and with unique identification all safety configuration items and safety information (hazards, safety requirements, risks, etc.)** **Level 3**

The safety configuration items and safety information required to achieve the safety integrity requirements of the safety-related system should be maintained accurately and with unique identification. A configuration item is an element designated for configuration management, which may consist of multiple related work products.

Supporting action(s)

- OS.STO.a5 Use of tools with support to cross reference and maintain the traceability among safety information in the software specification

PM.SCM.a2 **Define and document change-control procedures** **Level 3**

Change-control procedures and the strategy that will be adopted must be defined and recorded.

PM.SCM.a3 **Define and document safety configuration items to be included in the baseline** **Level 1**

The safety configuration items that will be included in the baseline should be defined and documented. Examples of criteria for selecting such items may be artifacts/information used by two or more groups, the ones that are expected to change over time either because of errors or change of requirements, dependent on each other and a change in one mandates a change in others and the ones critical for the project [28].

Supporting action(s)

- PM.SCM.a1 Maintain accurately and with unique identification all safety configuration items and safety information (hazards, safety requirements, risks, etc.)

PM.SCM.a4 Document configuration status, release status, the justification (taking account of the impact analysis) for and approval of all modifications, and the details of the modification **Level 3**

The configuration status, release status, the justification (taking account of the impact analysis) for an approval of all modifications, and the details of the modification should be recorded.

Supporting action(s)

- PM.SCM.a6 Perform safety impact analysis on changes

PM.SCM.a5 Document the release of safety-related software **Level 3**

The release of safety-related software, changes in the agreements with the suppliers, and other relevant information should be documented.

PM.SCM.a6 Perform safety impact analysis on changes **Level 2**

Change request may occur at any phase of the software safety lifecycle regarding artifacts or information specified earlier in the safety lifecycle. In this case, an impact analysis must be conducted to determine [38][45]: (1) which software modules are impacted; and (2) which earlier safety lifecycle activities shall be repeated.

Supporting action(s)

- PM.SCM.a7 Specify and follow the template for software modification request

PM.SCM.a7 Specify and follow the template for software modification request **Level 1**

A template for software modification request should be defined by the configuration management area and followed by all stakeholders of the organization.

PM.SCM.a8 Document the procedures for initiating modifications to the safety-related systems, and to obtain approval and authority for modifications **Level 2**

The procedures for initiating modifications to the safety-related systems, and to obtain approval and authority for modifications should be determined and recorded.

Supporting action(s)

- PM.SCM.a7 Specify and follow the template for software modification request

PM.SCM.a9 Maintain and make available the software configuration management log **Level 2**

A log with all commands executed in the artifacts, such as insertion, exclusion and update, must be maintained. This log must be accessible by all authorized stakeholder so they can be aware of all changes in such artifacts.

Supporting action(s)

- PM.SCM.a1 Maintain accurately and with unique identification all safety configuration items and safety information (hazards, safety requirements, risks, etc.)

PM.SCM.a10 Appoint all deliverable documents according to the rules defined in the Configuration Management Plan **Level 2**

A standard for naming the deliverable documents established in the configuration management plan should be followed.

Supporting action(s)

- PM.SCM.a1 Maintain accurately and with unique identification all safety configuration items and safety information (hazards, safety requirements, risks, etc.)

PM.SCM.a11 Upload all documents on the safety information system **Level 3**

The safety information system must be used to manage all documents produced during the development process.

PM.SCO Safety Communication

The safety analysis and assurance processes requires knowledge of many safety terms, methods, process from requirements engineers. However, they generally are unfamiliar with all such information. Aiming to minimize this problem, the safety module add actions to improve the safety communication sub process.

PM.SCO.a1 Establish formal communication channels among different organizational levels **Level 2**

Formal communication channels (for example email, face-to-face, meeting, collaboration infrastructure) among different organizational levels are also necessary to maintain continuous communication with internal stakeholders, including comprehensive reporting of safety performance.

Supporting action(s)

- OS.GSM.a8 Progress reports should be prepared in a period of time defined by the project

PM.SCO.a2 Define a method of exchanging safety information with the suppliers **Level 1**

Exchanging safety information with the suppliers is fundamental for the development of safety-critical systems. Therefore, adequate method for communication with suppliers must be defined.

PM.SCO.a3 Establish a common nomenclature **Level 1**

Common nomenclature is of paramount importance for specifying safety to avoid misunderstandings, redundancies and errors in system specification. Hence, the company should define a glossary and adopt at all levels of organization.

PM.SCO.a4 Train people continuously in system engineering and safety techniques (education) **Level 1**

Stakeholders should be trained continuously about methods, techniques, terms of system engineering and safety techniques to improve the safety analysis and the RE process.

PM.SCO.a5 Use of a common safety information system for system specification and safety analysis **Level 3**

The safety information should be shared with the purpose of specifying the system and conducting safety analysis. The use of a common system improves the communication among personnel improving the system safety.

PM.SCO.a6 Keep stakeholders updated regarding the progress of all safety-related activities **Level 3**

Stakeholders must be aware of the status of system development process. In order to achieve this, progress reports should be elaborated and published.

Supporting action(s)

- PM.SCO.a1 Establish formal communication channels among different organizational levels

PM.SCO.a7 Construct a repository of common hazards **Level 3**

A repository listing the common hazards can reduce the time spent in safety analysis contributing to a better analysis. Accordingly, such repository should be constructed and maintained.

PM.SCO.a8 Define and follow templates for system artifacts **Level 1**

Templates are important to optimize the specification, provide stakeholders with acquaintance about the artifacts and processes adopted by the company. Hence, templates for system artifacts must be established and followed.

PM.SCO.a9 Document how conflicts will be resolved **Level 1**

Misunderstandings and conflicts among safety goals or mission goals and safety goals for example may occur during system specification. Therefore, procedures to solve such conflicts must be established.

PM.SCO.a10 Identify, record and resolve conflicts **Level 1**

When conflicts are identified, they should be recorded and solved following the procedures defined previously.

Supporting action(s)

- PM.OS.GSM.a9 Document how conflicts will be resolved

PM.SCO.a11 Produce all the deliverables documents based on the official document templates **Level 2**

All deliverables documents should be produced according the templates defined by the company.

Supporting action(s)

- PM.SCO.a8 Define and follow templates for system artifacts

PM.SCO.a12 Make available safety-related software specification to every person involved in the lifecycle **Level 1**

The personnel involved in the system lifecycle must be able to visualize to the safety-related software specification with control access.

PM.ST Safety Traceability

Changes in requirements will probably occur during the system development. Therefore, it is necessary to ensure consistency among system artifacts. This sub process area of safety module handles the traceability among artifacts helping to determine that the requirements affected by the changes have been completely addressed.

PM.ST.a1 Define and maintain traceability policies **Level 3**

Traceability policies to be followed during the development process must be elaborated.

PM.ST.a2 Define and maintain bi-directional traceability between the system safety requirements and the software safety requirements **Level 3**

The safety-critical system is composed not only by software, hence, bi-directional traceability between the system safety requirements and the software safety requirements must be defined and maintained.

PM.ST.a3 Define and maintain bi-directional traceability between the safety requirements and the perceived safety needs **Level 3**

The relationships between the safety requirements and the perceived safety needs must be identified and maintained. If such relationships will be possible to determine which safety requirements satisfy some safety needs and vice-versa.

PM.ST.a4 Link and maintain bi-directional traceability between environmental assumptions and the parts of the hazard analysis based on the assumption **Level 3**

Environmental assumptions play an important role in safety analysis since their occurrence assumed by the requirements engineer may compromise the system safety. Hence, the links between the environmental assumptions and the parts of the hazard analysis based on the assumption must be properly maintained.

PM.ST.a5 **Define and maintain bi-directional traceability between system and subsystem verification results and system specification** **Level 3**

Bi-directional traceability between system and subsystem verification results and system specification must be established and maintained.

PM.ST.a6 **Define and maintain bi-directional traceability between validation results and system specification** **Level 3**

The relationships between the validation results and system specification must be established.

PM.ST.a7 **Define and maintain bi-directional traceability among system hazards into components** **Level 3**

The back and forth traceability between system hazards and its components must be defined and maintained.

PM.ST.a8 **Justify reasons for not traced software requirements** **Level 3**

The software requirements that are not traced must be documented and the reasons for such decision must be recorded.

2. Maturity Level View

In this section, the module can be viewed by maturity level (see Table 4). This view shows the practices from all process areas which the organization should implement in order to achieve a specific maturity level.

Table 4. Description of UNI-REPM safety module by maturity level view.

Level 1 - Basic	
RE	**Requirements Elicitation**
RE.SM	Supplier Management
RE.SM.a4	Specify all external systems and safety-related software
RE.SM.a5	Establish and maintain detailed system integration procedures for the external systems and safety-related software
RE.SM.a6	Define the safety standards that suppliers must follow
DS	**Documentation and Requirements Specification**
DS.HF	Human Factors
DS.HF.a2	Document human factors design and analysis
DS.HF.a3	Evaluate prototypes, requirements and technical UI restrictions
DS.SDO	Safety Documentation

DS.SDO.a2	Ensure that safety requirements are incorporated into system and subsystem specifications, including human-machine interface requirements
DS.SDO.a3	Document all lifecycle and modification activities
DS.SDO.a5	Document System Limitations
DS.SDO.a10	Include a summary of safety requirements
RA	**Requirements Analysis**
RA.PSA	Preliminary Safety Analysis
RA.PSA.a1	Identify and document safety-critical computer software components and units
RA.PSA.a3	Identify and document system hazards
RA.PSA.a4	Identify and document hazards, hazardous situations and harmful events due to interaction with other equipment or systems (installed or to be installed)
RA.PSA.a5	Specify the type of initiating events that need to be considered
RA.PSA.a6	Obtain and document information about the determined hazards (causes, probability, severity, duration, intensity, toxicity, exposure limit, mechanical force, explosive conditions, reactivity, flammability etc.)
RA.PSA.a7	Identify and document hazardous materials
RA.PSA.a8	Identify and document consequences of hazards, severity categories and affected assets
RA.PSA.a9	Conduct risk estimation
RA.PSA.a10	Conduct risk evaluation for each identified hazard
RA.PSA.a11	Identify and document risk mitigation procedures for each identified hazard
RA.PSA.a13	Identify and document pure safety requirements
RA.PSA.a14	Identify and document safety-significant requirements and safety integrity levels
RA.PSA.a15	Identify and document safety constraints and how they could be violated
RA.PSA.a16	Identify and document possible control flaws and inadequate control actions
RA.PSA.a17	Identify and document safety functional requirements
RA.PSA.a18	Identify and document operational requirements
RA.PSA.a21	Identify and document analysis and verification requirements, possible safety-interface problems, including the human-machine interface, and operating support requirements
RA.PSA.a23	Consolidate preliminary system safety technical specification
RA.FH	Failure Handling
RA.FH.a1	Define requirements for the avoidance of systematic faults
RA.FH.a2	Specify Fault-detection procedures
RA.FH.a3	Specify Restart-up procedures
RA.FH.a6	Perform reliability and system performance analysis
RP	**Release Planning**
RP.SC	Safety Certification
RP.SC.a2	Demonstrate the preliminary level of safety achieved by the system
RP.SC.a3	Evaluate the threat to society from the hazards that cannot be eliminated or avoided
RP.SC.a4	Construct preliminary safety and hazard reports
RP.SC.a5	Construct preliminary safety cases
RP.SC.a9	Specify a maintenance plan
RV	**Requirements Validation**

RV.SVV	Safety Validation and Verification
RV.SVV.a1	Define the safety validation plan for software aspects of system safety
RV.SVV.a2	Define the safety verification plan
RV.SVV.a6	Define and maintain a software integration test plan
RV.SVV.a9	Perform safety evaluation and verification at the system and subsystem levels
OS	**Organizational Support**
OS.SP	Safety Planning
OS.SP.a1	Develop an integrated system safety program plan
OS.SP.a3	Define and document the interface between system safety and all other applicable safety disciplines
OS.SP.a4	Delineate the scope of safety analysis
OS.SP.a5	Establish the hazards auditing and log file
OS.SP.a6	Establish working groups and structures
OS.SP.a7	Define and document the regulations and safety standards to be followed
OS.SP.a8	Identify any certification requirements for software, safety or warning devices or other special safety feature
OS.SP.a12	Specify operating conditions of the machine and installation conditions of the electronic parts
OS.SP.a13	Determine the required performance level
OS.SP.a14	Identify and document the hazard analysis to be performed; the analytical techniques (qualitative or quantitative) to be used; and depth within the system that each analytical technique will be used (e.g., system level, subsystem level, component level)
OS.GSM	General Safety Management
OS.GSM.a1	Identify and document the system development methodology
OS.GSM.a2	Identify and document safety lifecycle for the system development
OS.GSM.a3	Identify and document competence requirements for the safety activities
OS.GSM.a4	Set safety policy and define safety goals
OS.GSM.a5	Identify and document responsibility, accountability and authority
OS.GSM.a6	Define system safety program milestones and relate these to major program milestones, program element responsibility, and required inputs and outputs
PM	**Requirements Process Management**
PM.SCM	Safety Configuration Management
PM.SCM.a3	Define and document safety configuration items to be included in the baseline
PM.SCM.a7	Specify and follow the template for software modification request
PM.SCO	Safety Communication
PM.SCO.a2	Define a method of exchanging safety information with the suppliers
PM.SCO.a3	Establish a common nomenclature
PM.SCO.a4	Train people continuously in system engineering and safety techniques (education)
PM.SCO.a8	Define and follow templates for system artifacts
PM.SCO.a9	Document how conflicts will be resolved
PM.SCO.a10	Identify, record and resolve conflicts
PM.SCO.a12	Make available safety-related software specification to every person involved in the lifecycle

Level 2- Intermediate	
RE	**Requirements Elicitation**
RE.SM	Supplier Management
RE.SM.a1	Establish and maintain formal agreements among organization and suppliers
RE.SM.a2	Identify and document the products to be acquired
RE.SM.a3	Select suppliers and record rationale
DS	**Documentation and Requirements Specification**
DS.HF	Human Factors
DS.HF.a1	Construct operator task models
DS.HF.a4	Model and evaluate operator tasks and component black-box behavior
DS.HF.a5	Define interfaces considering ergonomic principles
DS.HF.a6	Specify Human Machine Interface requirements
DS.SDO	Safety Documentation
DS.SDO.a4	Develop and document training, operational and software user manuals
DS.SDO.a6	Provide a safety manual
DS.SDO.a7	Document lessons learned
DS.SDO.a8	Ensure that safety-related information is incorporated into user and maintenance documents
RA	**Requirements Analysis**
RA.PSA	Preliminary Safety Analysis
RA.PSA.a19	Perform and document the feasibility evaluation of safety functional requirements
RA.PSA.a20	Prioritize hazards and safety requirements
RA.PSA.a22	Perform interface analysis, including interfaces within subsystems (such as between safety-critical and non-safety-critical software components)
RA.FH	Failure Handling
RA.FH.a4	Document the system behavioral model
RA.FH.a5	Identify and document Common-cause failures (CCF) and how to prevent them
RP	**Release Planning**
RP.SC	Safety Certification
RP.SC.a1	Conduct safety audits
RP.SC.a6	Demonstrate preliminary compliance with safety standards
RP.SC.a8	Document the division of responsibility for system certification and compliance with safety standards during safety planning
RV	**Requirements Validation**
RV.SVV	Safety Validation and Verification
RV.SVV.a3	Define the technical strategy for the validation of external systems and safety-related software
RV.SVV.a4	Define pass/fail criteria for accomplishing software validation and verification
RV.SVV.a5	Develop safety test plans, test descriptions, test procedures, and validation and verification safety requirements
RV.SVV.a7	Validate safety-related software aspects
RV.SVV.a8	Ensure that there is no potentially hazardous control actions

RV.SVV.a10	Conduct joint reviews (company and customer)
RV.SVV.a11	Ensure that the stakeholders understand software-related system safety requirements and constraints
RV.SVV.a12	Document discrepancies between expected and actual results
RV.SVV.a13	Verify the behavioral model
RV.SVV.a14	Ensure that software requirements and interface specification are consistent
RV.SVV.a15	Perform safety inspections
RV.SVV.a16	Identify and fix inconsistencies safety requirements specification
OS	**Organizational Support**
OS.SP	Safety Planning
OS.SP.a2	Define and document requirements for periodic functional safety audits
OS.SP.a10	Review safety experience on similar systems
OS.GSM	General Safety Management
OS.GSM.a8	Prepare progress reports in a period of time defined by the project
OS.GSM.a9	Monitor project and take corrective actions
OS.STO	Safety Tool support
OS.STO.a1	Use of verification and validation tools
OS.STO.a4	Record information of the tools in the baseline
OS.STO.a6	Use of computer-aided specification tools
OS.SKM	Safety Knowledge Management
OS.SKM.a5	Document a strategy to manage the knowledge
OS.SKM.a6	Define a lifecycle for projects artifacts
PM	**Requirements Process Management**
PM.SCM	Safety Configuration Management
PM.SCM.a6	Perform safety impact analysis on changes
PM.SCM.a8	Document the procedures for initiating modifications to the safety-related systems, and to obtain approval and authority for modifications
PM.SCM.a9	Maintain and make available the software configuration management log
PM.SCM.a10	Appoint all deliverable documents according to the rules defined in the Configuration Management Plan
PM.SCO	Safety Communication
PM.SCO.a1	Establish formal communication channels among different organizational levels
PM.SCO.a11	Produce all the deliverables documents based on the official document templates

Level 3- Advanced	
DS	**Documentation and Requirements Specification**
DS.SDO	Safety Documentation
DS.SDO.a1	Record safety decisions and rationale
DS.SDO.a9	Maintain hazard and risk analysis results for the system throughout the overall safety lifecycle
RA	**Requirements Analysis**

RA.PSA	Preliminary Safety Analysis
RA.PSA.a2	Simulate the process
RA.PSA.a12	Collect safety requirements from multiple viewpoints
RP	**Release Planning**
RP.SC	Safety Certification
RP.SC.a7	Ensure that the hazard report is updated with embedded links to the resolution of each hazard, such as safety functional requirements, safety constraints, operational requirements, and system limitations
OS	**Organizational Support**
OS.SP	Safety Planning
OS.SP.a9	Define and document requirements completeness criteria and safety criteria
OS.SP.a11	Specify the general safety control structure
OS.GSM	General Safety Management
OS.GSM.a7	Use of indicators on engineering documentation to assess the product properties and the development progress
OS.STO	Safety Tool support
OS.STO.a2	Specify justifications for the selection of the off-line support tools
OS.STO.a3	Assess offline support tools which can directly or indirectly contribute to the executable code of the safety related system
OS.STO.a5	Use of tools with support to cross reference and maintain the traceability among safety information in the software specification
OS.STO.a7	Define and use tools to support the safety process and workflow management
OS.SKM	Safety Knowledge Management
OS.SKM.a1	Establish and maintain an infrastructure to share knowledge
OS.SKM.a2	Develop a safety information system to share knowledge in the organization
OS.SKM.a3	Define control access mechanisms to the safety information system
OS.SKM.a4	Maintain employees competence information
OS.SKM.a7	Define and maintain a strategy for reuse
OS.SKM.a8	Reuse the stored knowledge
OS.SKM.a9	Document the use of stored knowledge
OS.SKM.a10	Notify users about problems, new versions and exclusions of artifacts in use
OS.SKM.a11	Manage assets
PM	**Requirements Process Management**
PM.SCM	Safety Configuration Management
PM.SCM.a1	Maintain accurately and with unique identification all safety configuration items and safety information (hazards, safety requirements, risks, etc.)
PM.SCM.a2	Define and document change-control procedures
PM.SCM.a4	Document configuration status, release status, the justification (taking account of the impact analysis) for and approval of all modifications, and the details of the modification
PM.SCM.a5	Document the release of safety-related software
PM.SCM.a11	Upload all documents on the safety information system
PM.SCO	Safety Communication
PM.SCO.a5	Use of a common safety information system for system specification and safety analysis

PM.SCO.a6	Keep stakeholders updated regarding the progress of all safety-related activities
PM.SCO.a7	Construct a repository of common hazards
PM.ST	Safety Traceability
PM.ST.a1	Define and maintain traceability policies
PM.ST.a2	Define and maintain bi-directional traceability between the system safety requirements and the software safety requirements
PM.ST.a3	Define and maintain bi-directional traceability between the safety requirements and the perceived safety needs
PM.ST.a4	Link and maintain bi-directional traceability between environmental assumptions and the parts of the hazard analysis based on the assumption
PM.ST.a5	Define and maintain bi-directional traceability between system and subsystem verification results and system specification
PM.ST.a6	Define and maintain bi-directional traceability between validation results and system specification
PM.ST.a7	Define and maintain bi-directional traceability among system hazards into components
PM.ST.a8	Justify reasons for not traced software requirements

Glossary

Accident: an undesired and unplanned (but not necessarily unexpected) event that results in (at least) a specified level of loss (including loss of human life or injury, property damage, environmental pollution, and so on). In an insulin infusion pump, an accident can be *incorrect treatment received by the patient*.

Environmental conditions: the state of the environment. The set of factors including physical, cultural, demographic, economic, political, regulatory, or technological elements surrounding the system that could affect its safety. For example, in an insulin infusion pump, an environmental condition can be *obstruction in the delivery path*.

Harm: physical injury or damage to the health of people or damage to property or the environment.

Hazard: system state or set of conditions that, together with a particular set of worst-case environmental conditions, will lead to an accident (loss). One hazard in an insulin infusion pump can be an *insulin overdose.*

Pure safety requirements: are typically of the form of a quality criterion (a system-specific statement about the existence of a sub-factor of safety) combined with a minimum or maximum required threshold along some quality measure. They directly specify how safe the system must be. In an insulin infusion pump, *the difference between the programmed infusion and the delivered infusion shall not be greater than 0.5%*.

Safety-significant requirements: non-safety primary mission requirements, i. e. requirements that are not originally defined to mitigate some hazard, but they can have significant safety ramifications.

Safety functional requirements: Safety functional requirements are functions to be implemented in a safety-critical system that is intended to achieve or maintain a safe state for the system, in respect of a specific hazardous situation.

Safety Constraints: engineering decisions that have been chosen to be mandated as a requirement intended to ensure a minimum level of safety. Therefore, any safety-related or relevant constraints between the hardware and the software should be identified and documented.

Systematic faults: faults produced by human error during system development and operation that will always appear when the necessary environmental conditions occur.

Risk: combination of the probability of occurrence of a harm and its severity.

References

[1] A.C. Yeh, "Requirements Engineering Support Technique (REQUEST) A Market Driven Requirements Management Process," 1992, pp. 211-223.

[2] B. Regnell, P. Beremark, and O. Eklundh, "A market-driven requirements engineering process: results from an industrial process improvement programme," UK: Springer-Verlag, 1998, pp. 121-9.

[3] P. Carlshamre and B. Regnell, "Requirements lifecycle management and release planning in market-driven requirements engineering processes," Los Alamitos, CA, USA: IEEE Comput. Soc, 2000, pp. 961-5.

[4] R. Wieringa and C. Ebert, "RE'03: Practical requirements engineering solutions," IEEE Software, vol. 21, 2004, pp. 16-17.

[5] S. Fricker, T. Gorschek, C. Byman, A. Schmidle, "Handshaking with Implementation Proposals: Negotiating Requirements Understanding," IEEE Software, vol. 27, no. 2, pp. 72-80, Mar./Apr. 2010, doi:10.1109/MS.2009.195

[6] N.A.M. Maiden and G. Rugg, "ACRE: Selecting methods for requirements acquisition," Software Engineering Journal, vol. 11, 1996, pp. 183-192.

[7] CMMI for Development, Version 1.2, CMMI-DEV v1.2, CMU/SEI-2006-TR-008, Technical Report, Software Engineering Institute, August 2006

[8] D. Firesmith, "Prioritizing requirements," Journal of Object Technology, vol. 3, 2004, pp. 35-47.

[9] P. Carlshamre, K. Sandahl, M. Lindvall, B. Regnell, and J. Natt och Dag, "An industrial survey of requirements interdependencies in software product release planning," Los Alamitos, CA, USA: IEEE Comput. Soc, 2000, pp. 84-91.

[10] M. Khurum, K. Aslam, and T. Gorschek, "A method for early requirements triage and selection utilizing product strategies", Piscataway, NJ, USA: IEEE, 2008, pp. 97-104.

[11] T. Gorschek and C. Wohlin, "Requirements abstraction model," Requirements Engineering, vol. 11, 2006, pp. 79-101.

[12] Gorschek T., Tejle K., "A Method for Assessing Requirements Engineering Process Maturity in Software Projects", Blekinge Institute of Technology, Master Thesis Computer Science no. MSC-2002:2, 2002.

[13] L. Karlsson and B. Regnell, "Introducing tool support for retrospective analysis of release planning decisions," Berlin, Germany: Springer-Verlag, 2006, pp. 19-33.

[14] P. Sawyer, I. Sommerville, and G. Kotonya, "Improving market-driven RE processes," Espoo, Finland: Tech. Res. Centre Finland, 1999, pp. 222-36.

[15] M. Khurum, T. Gorschek, M. Wilson "The software value map—an exhaustive collection of value aspects for the development of software intensive products", Journal of Software: Evolution and Process vol. 25, 2013, pp. 711-741.

[16] Jeffery R. Value-Based Software Engineering. Springer: Germany, 2006.

[17] I. Sommerville, P. Sawyer, "Requirements Engineering: A Good Practice Guide", John Wiley & Sons, 1997.

[18] J. Natt och Dag, V. Gervasi, S. Brinkkemper, B. Regnell, ” Speeding up requirements management in a product software company: linking customer wishes to product requirements through linguistic engineering”, 12th Int. IEEE Requirements Engineering Conference, 2004, pp. 283-294.

[19] R. Grammes, R. Gotzhein. "SDL Profiles – Formal Semantics and Tool Support". Springer.

[20] Bowen, “Formal Specification and Documentation using Z: A Case Study Approach.” International Thomson Computer Press. ISBN 1-85032-230-9, 1996

[21] M. Hennessy: Algebraic Theory of Processes, The MIT Press, ISBN 0-262-08171-7.

[22] A. van Lamsweerde, E. Letier. ”From Object Orientation to Goal Orientation: A Paradigm Shift for Requirements Engineering. Proc. Radical Innovations of Software and Systems Engineering, LNCS, 2003.

[23] E. Yu, “Towards Modelling and Reasoning Support for Early-Phase Requirement Engineering”, IEEE International Symposium on Requirements Engineering, 1997, pp. 226 - 235.

[24] B. Regnell and S. Brinkkemper “Market-Driven Requirements Engineering for Software Products” in C. Wohlin and A. Aurum “Engineering and Managing Software Requirements”, Springer 2005.

[25] G. Ruhe “Product Release Planning: Methods, Tools and Applications” CBC Press, 2013.

[26] The ISPMA Foundational Level Syllabus version 1.2 http://ispma.org/wp-content/uploads/2014/02/ISPMA-SPM-FL-Syllabus-V.1.2.pdf accessed 27.11.2014

[27] S. Fricker, T. Gorschek, C. Byman, A. Schmidle, “Handshaking with Implementation Proposals: Negotiating Requirements Understanding”, IEEE Software, Vol. 27, No. 2 March/April 2010.

[28] CMMI. Capability maturity model® integration for Systems Engineering/Software Engineering (CMMI-SE/SW). Version 1.02, 2000.

[29] B. Edwards. Best Safety Practices Now and in the Future. In: Pharmacovigilance, Springer International Publishing, 2017. pp. 35-48.

[30] N. Leveson, Nancy. Engineering a safer world: Systems thinking applied to safety. Mit Press, 2011.

[31] T. Kontogiannis; M. C. Leva; N. Balfe. Total Safety Management: Principles, processes and methods. Safety Science, 2016.

[32] Departament of Defense of United States of America. MIL-STD-882D: Military standard - standard practice for system safety.

[33] ISO/IEC, International Organization for Standardization and International electrotechnical commission. ISO 15998: Earth-moving machinery - machine-control systems (mcs) using electronic components - performance criteria and tests for functional safety.

[34] D. Firesmith. Engineering safety-related requirements for software-intensive systems. In: Proceedings of the 28th international conference on Software engineering, ACM, 2006. pp. 1047-1048.

[35] K. Kazaras.; K. Kirytopoulos. Applying stamp in road tunnels hazard analysis. In: IET Conference Proceedings, The Institution of Engineering & Technology, 2011.

[36] K. Cox; M. Niazi; J. Verner. Empirical study of Sommerville and Sawyer's requirements engineering practices. In: IET software, v. 3, n. 5, 2009, pp. 339-355.

[37] N. Leveson. SAFEWARE: system safety and requirements. Addison-Wesley: 1995.

[38] ISO/IEC, International Organization for Standardization and International electrotechnical commission. ISO 61508: Functional Safety of Electrical/Electronic/Programmable Electronic Systems.

[39] Department of Defense of United States of America, MIL-STD-882C: Military standard - system safety program requirements.

[40] ISO. 13849-2: Safety of machinery - Safety related parts of control systems. Part 2: Validation (ISO 13849), v. 2, 2003.

[41] E. C. for Space Standardization, ECSS-E-HB-40A: Space engineering - software engineering handbook, ESA Requirements and Standards Division.

[42] E. C. for Space Standardization, ECSS-E-ST-40C: Space engineering - software, ESA Requirements and Standards Division.

[43] A. Saeed, R. de Lemos, and T. Anderson. On the safety analysis of requirements specifications for safety-critical software. In: ISA Transactions, vol. 34, no. 3, 1995, pp. 283-295.

[44] Department of Defense of United States of America, MIL-STD-882E: Military standard - system safety.

[45] ISO/IEC, International Organization for Standardization and International electrotechnical commission. ISO 15504-10: Information technology - Process assessment - Part 10: Safety extension.

[46] Somerville I., "Software Engineering", Addison-Wesley, 1995.

[47] Wohlin C., Aurum A., "Engineering and Managing Software Requirements", Springer, 2005.

[48] Karlsson, L., Dahlstedt, A.G., Regnell, B., Natt och Dag, J., Persson, Requirements engineering challenges in market-driven software development - An interview study with practitioners, In the Journal of Information and Software Technology, 49, 6, pp. 588-604, 2007.

[49] Regnell B., Beremark P., and Eklundh O., "A Market-driven Requirements Engineering Process - Results from an Industrial Process Improvement Programme", Springer, pp. 121-129, 1998.

[50] Juristo N., Moreno A.M, Silva A., "Is the European Industry Moving Toward Solving Requirements Engineering Problems?", IEEE Software, vol.19, no.6, pp. 70-77, Nov/Dec 2002.

[51] Beecham, S., Hall, T., & Rainer, A., Software process problems in twelve software companies: An empirical analysis, Empirical Software Engineering, 8, 7–42, 2003

[52] Niazi, M., An empirical study for the improvement of requirements engineering process, The 17th International Conference on Software Engineering and Knowledge Engineering , pp. 396–399, 2005.

[53] Hall T., Beecham S., Rainer A., "Requirements Problems in Twelve Companies: An Empirical Analysis", IEE Proceedings for Software, October, vol.149, no.5, pp.153-160, 2002.

[54] Gorschek T., "Requirements Engineering Supporting Technical Product Management", Karlskrona : Blekinge Institute of Technology, 2006.

[55] Villalón C., Agustín C., Gilabert S., Seco D., Sánchez G., and Cota P., "Experiences in the Application of Software Process Improvement in SMES," Software Quality Journal, vol. 10, pp. 261 – 273, 2002.

[56] Paulk M.C., Curtis B., Chrissis M.B. and Weber C., Capability Maturity Model TM for Software Version 1.1, 1993.

[57] CMMI for Development, Version 1.2, CMMI-DEV v1.2, CMU/SEI-2006-TR-008, Technical Report, Software Engineering Institute, August 2006, URL: http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf.

[58] The TickIT Guide – Using ISO 9001:2000 for Software Quality Management System, Construction, Certification and Continual Improvement, Issue 5.0, 2001.

[59] Ian Somerville and Pete Sawyer, Requirements Engineering – A Good Practice Guide, John Wiley & Sons, Chichester UK, 2000.

[60] Gorschek T., Tejle K., "A Method for Assessing Requirements Engineering Process Maturity in Software Projects", Blekinge Institute of Technology, Master Thesis Computer Science no. MSC-2002:2, 2002.

[61] Wiegers K., Software requirements: Practical techniques for gathering and managing requirement through the product development cycle, Microsoft Press. Redmond, Washington, 2003.

[62] M. Svahnberg, T. Gorschek, T. T. L. Nguyen, and M. Nguyen, "Unirepm: a framework for requirements engineering process assessment," Requirements Engineering, vol. 20, no. 1, pp. 91–118, 2015.

[63] M. Svahnberg, T. Gorschek, T. T. L. Nguyen, and M. Nguyen, "Unirepm: validated and improved," Requirements Engineering, vol. 18, no. 1, pp. 85–103, 2013.

[64] P. Sawyer, I. Sommerville, and S. Viller, "Requirements process improvement through the phased introduction of good practice," Software Process: Improvement and Practice, vol. 3, no. 1, pp. 19–34, 1997.

[65] T. Gorschek, M. Svahnberg, and K. Tejle, "Introduction and application of a lightweight requirements engineering process," in Ninth International Workshop on Requirements Engineering: Foundation for Software Quality, 2003.

[66] T. Gorschek, A. Gomes, A. Pettersson, and R. Torkar, "Introduction of a process maturity model for market-driven product management and requirements engineering," Journal of software: Evolution and Process, vol. 24, no. 1, pp. 83–113, 2012.

[67] T. L. Reis, M. A. S. Mathias, and O. J. de Oliveira, “Maturity models: identifying the state-of-the-art and the scientific gaps from a bibliometric study,” Scientometrics, pp. 1–30, 2016.

[68] N. Leveson, Engineering a safer world: Systems thinking applied to safety. Mit Press, 2011.

[69] J. Vilela, J. Castro, L. E. G. Martins, and T. Gorschek, “Integration between requirements engineering and safety analysis: A systematic literature review,” Journal of Systems and Software, 2016.

[70] N. G. Leveson, Safeware: system safety and computers. ACM, 1995.

[71] R. R. Lutz, “Software engineering for safety: a roadmap,” in Proceedings of the Conference on The Future of Software Engineering. ACM, 2000, pp. 213–226.

[72] R. Guillerm, H. Demmou, and N. Sadou, “Information model for model driven safety requirements management of complex systems,” in Complex Systems Design & Management. Springer, 2010, pp. 99–111.

[73] A. Simpson and J. Stoker, “Will it be safe??an approach to engineering safety requirements,” in Components of System Safety. Springer, 2002, pp. 140–164.

[74] N. G. Leveson, “An approach to designing safe embedded software,” in Embedded Software. Springer, 2002, pp. 15–29.

[75] K. Cox, M. Niazi, and J. Verner, “Empirical study of sommerville and sawyer’s requirements engineering practices,” IET software, vol. 3, no. 5, pp. 339–355, 2009.

[76] R. B. Ahmad, M. H. N. M. Nasir, J. Iqbal, and S. M. Zahid, “High perceived-value requirements engineering practices for outsourced software projects.” JSW, vol. 10, no. 10, pp. 1199–1215, 2015.

[77] B. Solemon, S. Sahibuddin, and A. A. A. Ghani, “Requirements engineering problems in 63 software companies in malaysia,” in Information Technology, 2008. ITSim 2008. International Symposium on, vol. 4. IEEE, 2008, pp. 1–6.

[78] D. Firesmith, “Engineering safety-related requirements for software-intensive systems,” in Proceedings of the 28th international conference on Software engineering. ACM, 2006, pp. 1047–1048.

[79] P. Panaroni, G. Sartori, F. Fabbrini, M. Fusani, and G. Lami, “Safety in automotive software: an overview of current practices,” in Computer Software and Applications, 2008. COMPSAC’08. 32nd Annual IEEE International. IEEE, 2008, pp. 1053–1058.

[80] P. J. Graydon and C. M. Holloway, “Planning the unplanned experiment: Assessing the efficacy of standards for safety critical software,” 2015.

[81] L. E. G. Martins and T. Gorschek, “Requirements engineering for safety-critical systems: A systematic literature review,” Information and Software Technology, vol. 75, pp. 71–89, 2016.

[82] B. Solemon, S. Sahibuddin, and A. A. A. Ghani, “Requirements engineering problems and practices in software companies: An industrial survey,” in International Conference on Advanced Software Engineering and Its Applications. Springer, 2009, pp. 70–77.

[83] M. Lubars, C. Potts, and C. Richter, “A review of the state of the practice in requirements modeling,” in Proceedings of IEEE International Symposium on Requirements Engineering. IEEE, 1993, pp. 2–14.

[84] U. Nikula, J. Sajaniemi, and H. Kalviainen, A State-of-the-practice Survey on Requirements Engineering in Small-and Medium-sized Enterprises. Lappeenranta University of Technology Lappeenranta, Finland, 2000.