Maturity	Process Owner	Check	Release	Description
Valid	Name / Department	Name / Department	Name / Department	Detailed procedure for software development

Title: **Software Development Procedure**

Purpose:

This document describes the overall software development process of microcontroller software during all phases of the Company Name product life cycle.

Scope:

Valid for all software development in Company Name

TABLE OF CONTENTS

TABLE OF FIGURES

1. Terminology, Definitions and Abbreviations

Term / Abbreviation	Denotation
SW	Software
SW-CM	Software Configuration Manager
SW-D	Software Developer
SW-Management	Software Group Leader or Domain Leader
SW-RE	Software Requirements Engineer
SW-PM	Software Project Manager
SW-QE	Software Quality Engineer
SW-T	Software Tester
SW-TM	Software Test Manager
HW	Hardware
TBD	To be defined

2. Process

2.1 General Remarks

All outputs named in the following detailed procedure have to be under configuration management, even if this is not explicitly stated in each process step.

Please observe: For many of the work products (documents) an object review is mentioned in this process. You have to tailor this according to your needs. It is usually not possible and not necessary to perform an object review on each small modification of a document. It is also not necessary to perform a code review on each software unit and on each modification of a software unit. SPICE requires from you to have a strategy and to document this strategy. E.g. only review the initial versions of a document and only review the software units with high criticality. Document this in the quality assurance section of the project plan and delete or rephrase the passages in this document.

2.2 Process Workflow

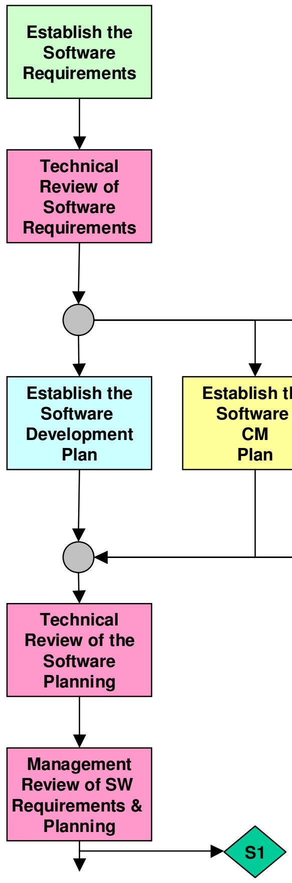
Establish the Software Requirements
Technical Review of Software Requirements
Establish the Software Development Plan
Establish the Software CM Plan
Establish the Software Quality Assurance Plan
Establish the Software Test Plan
Technical Review of the Software Planning
Management Review of SW Requirements & Planning
S1

Figure 1: Software Development until Milestone S1

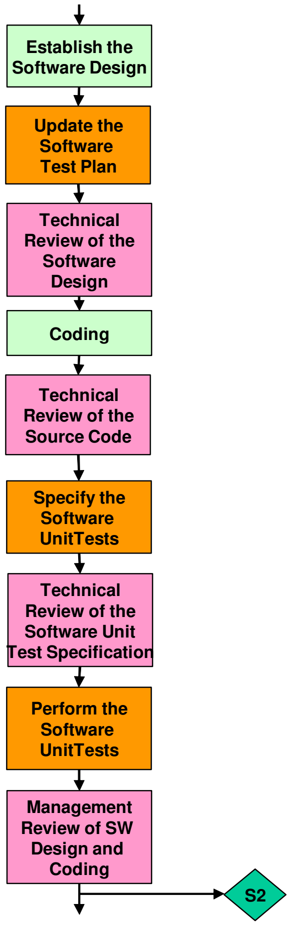
Establish the Software Design
Update the Software Test Plan
Technical Review of the Software Design
Coding
Technical Review of the Source Code
Specify the Software UnitTests
Technical Review of the Software Unit Test Specification
Perform the Software UnitTests
Management Review of SW Design and Coding
S2

Figure 2: Software Development until Milestone S2

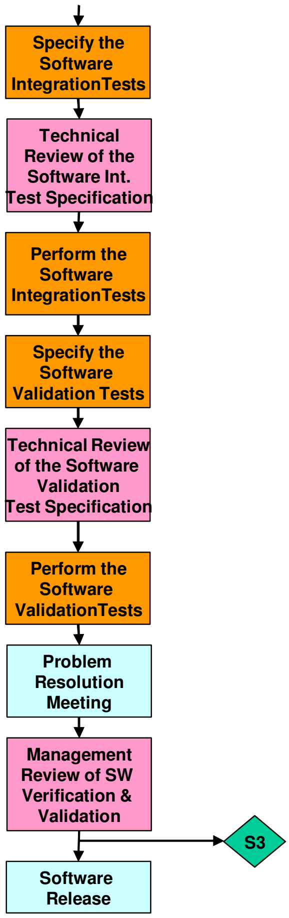
Specify the Software IntegrationTests
Technical Review of the Software Int. Test Specification
Perform the Software IntegrationTests
Specify the Software Validation Tests
Technical Review of the Software Validation Test Specification
Perform the Software ValidationTests
Problem Resolution Meeting
Management Review of SW Verification & Validation
S3
Software Release

Figure 3: Software Development until the Software Release

2.3 Process Description

2.3.1 General

Describe here some general things. E.g. how the release loops are handled, e.g. if it is possible to exit at any stage of the process and enter at a previous stage i.e. to fix problems prior to a moving on in the process. If your software development is part of a system development process describe how the process is embedded into the overall process and how the interfaces are managed.

2.3.2 Software Development Process

2.3.2.1 Establish the Software Requirements

Goal:	**Establish a valid Software Requirements Specification** - Software Requirements are identified and specified. - Deviations from the Stakeholder Requirements are identified and reconciled with the customer / stakeholder
Input:	Customer or other Stakeholder Requirements
Output:	Valid document: SoftwareRequirementsSpecification.doc
Methods and Templates:	Software Requirements Engineering Method SoftwareRequirementsSpecificationTemplate.doc

Action	**Action Description**
Analyze stakeholder requirement documents	• Derive Software requirements applying the methods as provided in the section "Methods and Templates:". • Note down open issues and assumptions for later discussion with the stakeholder. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-RE
Specify additional requirements	Generally, after analyzing the stakeholder requirements some requirements have not been covered or are not complete. • Not yet covered or deviating stakeholder requirements from the previous action have to be clarified with the stakeholder and included into the specifications. • Perform a proper use case analysis, considering all possible users of the system and generate scenarios for each use case to make sure that the requirements cover all needed topics. E.g. development or manufacturing needs are very often neglected in the requirements. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-RE

Action	Action Description
Clarify open issues and assumptions	• Clarify the Open Issues and assumptions from the use case analysis with the stakeholders. • Update the requirements as needed • Make sure that the performed updates do not lead to contradictions. If this should happen, solve contradictions. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-RE
Collect feedback on requirements from all parties concerned	• Contact all parties which are concerned by the requirements (e.g. test management, manufacturing, architects, representatives of the next development level / step) and collect their feedback regarding e.g. testability, feasibility, manufacturability. • Problems for individual requirements should be reconciled and fixed in the software requirements specification. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs.. **Responsible:** SW-PM **Participants:** SW-RE

2.3.2.2 Perform a Technical Review on the Software Requirements

Goal:	**Establish a reviewed version of the work product** The goal is to review the completed version of a work product and to evaluate its suitability for the intended use.
Input:	Completed version of the work product to be reviewed
Output:	Reviewed version of the work product Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	Action Description
Plan and initiate the technical review	This description is valid for all technical reviews in the software development. The technical review (document review) has to be performed on the work product in scope. A dedicated meeting for a walk through is not mandatory. The reviews can be performed as peer reviews. The owner or responsible of the object to be reviewed has to distribute the review object to appropriate peer reviewers. It is mandatory to have at least two peer reviewers. Further the owner of the object has to set the date of the review, i.e. when he expects the review results to be back. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM

Action	Action Description
	Participants: none
Preparation by the reviewers	The peer reviewers will carefully study the review object. The main focus is to answer the related review checklist questions. If a question is not to be ticked as o.k. a comment has to be filled-in in the comment sheet. Further the peer reviewer has to apply his experience and common sense to detect problems and errors in the reviewed object which may not be covered by the review checklist questions. These have also to be reported in the comments. The prepared checklists and comment sheets have to be ready for the review meeting or alternatively have to be handed in to the responsible of the technical review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T, SW-PM depending on the scope of the review.
Perform the technical review meeting	The responsible of the technical review has to call a review meeting where the peer reviewers participate to reconcile and discuss the review findings. Alternatively he can do this without any participants, using the pre-filled comments and checklists of the peer reviewers. The responsible for the review has to set a status of the review i.e. "passed" (Finished) or "failed" (Not finished). The peer reviewers have to agree to this report by their signature. If the review status was "passed" (Finished) and the need for rework was not identified there are no further actions required. If the modifications to the object are only minor, an agreement has to be reached among the reviewers and the responsible, about what should be modified and the owner or responsible of the object has to rework without new review. If the status of the technical review was "failed" (Not finished), the responsible of the technical review has to modify the review object according to the findings of the review. This may involve discussions and agreements with the peer reviewers until a satisfactory solution is found and implemented. The goal is that the peer reviewers can agree to the modification. This means that the technical review has to be performed again on the modified review object and the described procedure has to be followed until the status can be set to "passed" (Finished). **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T depending on the scope of the review.
Documentation and tracking	The review results and related forms and documents. I.e. the review question list, comment list and the status result, have to be placed into the project CM archives. The labeling of theses files has to be performed to relate them to the appropriate baseline. All review findings which should lead to a modification of the review object have to be submitted to the problem resolution process. I.e. for the

Action	Action Description
	findings appropriate entries have to be made in a tracker database or the Action Item List of the project. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** none

2.3.2.3 Initiate and Plan the Software Development

Goal:	**Establish the Software Development Planning** The roles and responsibilities within the project have to be clearly defined at this point. The Software Development Plan has to be established. Note that this step is in parallel to the steps to initiate and plan the configuration management and to initiate and plan the quality assurance.
Input:	Software Requirements Specification in sufficient quality to allow resource estimations. **Note:** Requirements engineering and planning of the SW development are not sequential. They run in parallel. Some of the requirements engineering is needed to perform the planning and the bigger part of the requirements engineering needs to be planned and documented in the Software Development Plan.
Output:	Valid document: SoftwareDevelopmentPlan.doc
Methods and Templates:	Software Project Planning Method SoftwareDevelopmentPlanTemplate.doc MeetingMinutesTemplate.doc ActionItemListTemplate.doc SoftwareResourceEstimationSheetTemplate.doc Software Project Time Schedule (e.g. in mpp format)

Action	Action Description
Nominate the SW-PM	Nominate the Software Project Manager for the project the latest at this point in time. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-Management **Participants:** SW-PM
Nominate the SW team members	Nominate the members of the software team, i.e. the developers. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-Management **Participants:** SW-PM, SW-D

Action	Action Description
Generate the Software Development Plan	Generate the Software Development Plan using the template SoftwareDevelopmentPlanTemplate.doc. The Plan should address the following items: Project Planning: Tailoring of the procedures, risk management, development goals, deliveries and milestones, organization and responsibilities in the project. The detailed schedule should be not included here, but kept in an appropriate tool. For resource estimation the template SoftwareResourceEstimationSheet.doc has to be used. This resource estimation has to be the base for the project schedule. Project Control: Specify which measures are applied to supervise the project status and perform reporting. Testing: Usually only a reference to an external Software Test Plan is made. Testing can only be established in its complete version after a valid Software Design Document. Therefore it is not required at this point in time. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM Participants: none
Perform the continuous project planning and tracking activities	Perform the continuous project planning and tracking as described in the "Software Project Planning Method". This involves especially the frequent update of the time schedule. Further there should be a project meeting established where the tracking of the project is done. The meeting minutes according to template MeetingMinutesTemplate.doc should be filled in as a documentation of the tracking. For small projects where a regular meeting seems inappropriate e.g. if only one person is assigned to the project the action item list according to template ActionItemListTemplate.doc can be used instead. **Possible Tailoring**: describe here in which cases and how the process can be tailored according to project needs. **Responsible**: SW-PM **Participants**: none

2.3.2.4 Initiate and Plan the Software Configuration Management

Goal:	**Establish the Configuration Management for Software Development** The role of a configuration manager within the project has to be assigned at this point. The Software Configuration Management Plan has to be established for project and the infrastructure for CM has to be set up, as e.g. archives and tools.
Input:	Software Development Plan (partially filled in)
Output:	Valid document: SoftwareConfigurationManagementPlan.doc and CM infrastructure
Methods and	Software Configuration Management Method

Templates:	SoftwareConfigurationManagementPlanTemplate.doc

Action	**Action Description**
Nominate a SW-CM	Nominate a Configuration Manager for the project to manage and / or perform all necessary CM activities. This role may be recruited from the software developers, although it is recommended to have a dedicated SW-CM. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-CM
Generate the Software CM Plan	Change management, version control, communication, data spaces and structures, tools and their maintenance have to be planned and set up. The planning has to be documented in a separate CM plan. Use the CM plan template "SoftwareConfigurationManagementPlanTemplate.doc " for this purpose. **Note**: it is possible and recommendable to perform CM not on project but on organization level. I.e. the archive structures, labeling and naming conventions have to be the same for all projects. In this case it is also recommendable to have a generic CM plan. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-CM
Setup the CM spaces	Change management, version control, communication, data spaces and structures, tools and their maintenance have to set up as described in the “Software Configuration Management Method" and the "SoftwareConfigurationManagementPlan.doc". **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-CM
Perform the continuous CM activities	Perform the continuous activities of CM as described in the "Software Configuration Management Method". **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-CM Participants: none

2.3.2.5 Establish the Software Test Plan

Goal:	**Establish the Software Test Plan** The scope of the SW Test Plan is: • Software Unit Test • Software Integration and Validation Test Goals of the Software Unit Test: • Approval that the software units are compliant to the Software

	Design Document, as well as the discipline related programming standards and rules. Goals of the Software Integration and Validation Test: • Approval that the different software units or components interact correctly at their interfaces as defined in the software architecture. This includes possible dynamic behavior and timing. • Approval that the software meets the requirements as defined in the Software Requirements Specification • To achieve these goals, this task provides the basis from a test management point of view to focus on appropriate (e.g. high priority) test goals and test topics Realistic effort / costs estimations and scheduling for related work packages to enable their systematic performance. Note, that the Test Plan shall be established in parallel to the software project planning, based in the software requirements. However, it needs to be updated after the design is established to reflect all architecture and design details.
Input:	Software Requirements Specification Software Development Plan Software Design Document
Output:	SoftwareTestPlan.doc
Methods and Templates:	Software Testing Method SoftwareTestPlanTemplate.doc

Action	**Action Description**
Generate the Software Test Plan	Identify the test requirements and define the test strategy and focus for the project, incl.: • Test goals • Test phases, • Test objects • Test methods, • Test documentation • Test end criteria Define the integration approach and integration steps. Define the test methods, tools and environment e.g. development environment, test stubs, conditional compilation, language dependent solutions, test equipment, specific skills and trainings of the testers, etc. Clarify the management of anomalies, i.e. reporting and tracking Estimate the effort, costs and schedules of test engineering activities, including: • Test phases and -activities (resources, schedules, investments, training) • Detailed list of test objects • Development of test specifications • Provision of test environment / test data / test programs • Training • Test conduct

	• Documentation of tests • Review of results Clarify the responsibilities for test engineering activities and synchronize test engineering activities with other plans e.g.: • System Development Plan • Overall Test Plan (Software Validation as part of Product Test) Document the above steps in the Software Test Plan. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-TM, SW-T

2.3.2.6 Initiate and Plan the Software Quality Assurance

Goal:	**Establish the Quality Assurance Plan for Software Development** The role of a quality planning engineer for the project has to be assigned at this point. The Software Development Plan contains a chapter about quality assurance. This has to be filled in.
Input:	Software Development Plan (partially filled in)
Output:	Software Development Plan with the part of the quality assurance filled in.
Methods and Templates:	Software Project Planning Method

Action	**Action Description**
Nominate the SW-QE	Nominate the Software Quality Engineer for the project to manage and / or perform all quality activities. It is mandatory that the person fulfilling this role is not member of the project development team. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** Q-Management **Participants:** SW-Management, SW-PM, SW-QE
Generate the Software Quality Assurance Plan	The SW-QE has to define the quality assurance measures applied in the project in agreement with the SW-PM, (these are e.g. the reviews, project control, status reporting and metrics). This is pre-defined in the section about quality assurance in the software development plan and may be tailored according to the allowed tailoring measures. No separate review is defined for the quality status section. It is the reviewed together with the review of the complete Software Development Plan. **Possible Tailoring:** none **Responsible:** SW-QE **Participants:** SW-PM

Action	Action Description
Perform the continuous QA activities	Perform the continuous quality assurance activities. This involves a regular evaluation of the status of a project by independent quality assurance staff. The quality engineer has to document the status in the Software Quality Status Sheet and report it in a defined frequency to the SW management. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-QE **Participants:** none

2.3.2.7 Perform a Technical Review on the Software Planning

Goal:	**Establish a reviewed version of the work product** The goal is to review the completed version of a work product and to evaluate its suitability for the intended use.
Input:	Completed version of the work product to be reviewed
Output:	Reviewed version of the work product Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	Action Description
Plan and initiate the technical review	This description is valid for all technical reviews in the software development. The technical review (document review) has to be performed on the work product in scope. A dedicated meeting for a walk through is not mandatory. The reviews can be performed as peer reviews. The owner or responsible of the object to be reviewed has to distribute the review object to appropriate peer reviewers. It is mandatory to have at least two peer reviewers. Further the owner of the object has to set the date of the review, i.e. when he expects the review results to be back. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** none
Preparation by the reviewers	The peer reviewers will carefully study the review object. The main focus is to answer the related review checklist questions. If a question is not to be ticked as o.k. a comment has to be filled-in in the comment sheet. Further the peer reviewer has to apply his experience and common sense to detect problems and errors in the reviewed object which may not be covered by the review checklist questions. These have also to be reported in the comments. The prepared checklists and comment sheets have to be ready for the review meeting or alternatively have to be handed in to the responsible of the technical review.

Action	Action Description
	Possible Tailoring: describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T, SW-PM depending on the scope of the review.
Perform the technical review meeting	The responsible of the technical review has to call a review meeting where the peer reviewers participate to reconcile and discuss the review findings. Alternatively he can do this without any participants, using the pre-filled comments and checklists of the peer reviewers. The responsible for the review has to set a status of the review i.e. "passed" (Finished) or "failed" (Not finished). The peer reviewers have to agree to this report by their signature. If the review status was "passed" (Finished) and the need for rework was not identified there are no further actions required. If the modifications to the object are only minor, an agreement has to be reached among the reviewers and the responsible, about what should be modified and the owner or responsible of the object has to rework without new review. If the status of the technical review was "failed" (Not finished), the responsible of the technical review has to modify the review object according to the findings of the review. This may involve discussions and agreements with the peer reviewers until a satisfactory solution is found and implemented. The goal is that the peer reviewers can agree to the modification. This means that the technical review has to be performed again on the modified review object and the described procedure has to be followed until the status can be set to "passed" (Finished). **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T depending on the scope of the review.
Documentation and tracking	The review results and related forms and documents. I.e. the review question list, comment list and the status result, have to be placed into the project CM archives. The labeling of theses files has to be performed to relate them to the appropriate baseline. All review findings which should lead to a modification of the review object have to be submitted to the problem resolution process. I.e. for the findings appropriate entries have to be made in a tracker database or the Action Item List of the project. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** none

2.3.2.8 Perform a S1 Management Review

Goal:	**Generate an agreement to enter the next development phase** The goal is to evaluate the completed work products of the previous development phase and to determine their suitability for the next development steps. This description is valid for all management reviews in the software development.
Input:	Completed versions of the work products of the previous development phase
Output:	Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	Action Description
Plan and initiate the management review	The management review has to be performed on the project progress of the project in scope. The SW-PM has to distribute the documents of the previous development phase to the review participants and allow enough time for preparation between the distribution and the review meeting. The SW-PM has to set the date of the review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** none
Perform the management review meeting	The SW-PM has to call a review meeting where the project progress will be evaluated. The focus in the meeting is to confirm that the work products of the previous development phase are present, technically reviewed and of sufficient quality to continue with the next step in the software development process. Further, the scope is to check on the project performance concerning the schedule, resources and quality. The participants of the review have to set a status of the review i.e. "passed" or "failed". The participants have to agree to this report by their signature. If the review status was "passed" there is the possibility that further actions are required but without a need of a new review. If the required activities in the project are only minor, an agreement has to be reached among the reviewers, about required activities and their tracking. If the status of the management review was "failed", the SW-PM has to solve the found problems and call for a repeated review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-Management, SW-D, SW-T depending on the scope of the review.

2.3.2.9 Milestone S1

This milestone marks the end of the requirements and planning phase for the software sub-project and the start of the detailed design phase. The successful passing of the S1 management review constitutes automatically the reaching of the S1 milestone.

2.3.2.10 Establish the Software Design

Goal:	**Establish the Software Design Document** The goal is to establish the high level software architecture and design and to establish the detailed unit design for the project.
Input:	Valid Software Requirements Specification.
Output:	Valid document: SoftwareDesignDescription.doc
Methods and Templates:	Software Design Method SoftwareDesignDescriptionTemplate.doc

Action	**Action Description**
Generate the Software Design Document	Use the valid Software Requirements Specification as well as the selected design methods and generate the high level design (architecture). Use the template SoftwareDesignDescriptionTemplate.doc and generate the Software Design Document containing the high level design of the new software. It has to show the important interfaces and functionalities. The modularization into functional blocks and the related control and data flow have to be documented. Also establish the detailed unit design in the Software Design Document. The software tester has to be consulted to make sure that their needs are reflected in the design. Further, the Software Design Method has to be followed for the design. The use of a design tools is permissible as long as the documentation (software design document) is established in sufficient quality. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D **Participants:** SW-T, SW-D
Generate the Prototype Source Code	Implement the source code of the new software complying to the high level design and detailed unit design of the Software Design Document. This may be the first generation of the units, to provide the empty software unit structure and development environment. It may come from a design tool as automatically generated code. For the implementation of the source code the relevant source code templates have to be used. Further the Software Design Method has to be followed for the implementation. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D **Participants:** none

2.3.2.11 Update the Software Test Plan

Make sure that the Software Test Plan is updated after the design is established. At this point the module break down was performed and the detailed test planning can be performed. Make sure that the updated test plan is reviewed again.

2.3.2.12 Perform a Technical Review on the Software Design

Goal:	**Establish a reviewed version of the work product** The goal is to review the completed version of a work product and to evaluate its suitability for the intended use.
Input:	Completed version of the work product to be reviewed
Output:	Reviewed version of the work product Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	Action Description
Plan and initiate the technical review	This description is valid for all technical reviews in the software development. The technical review (document review) has to be performed on the work product in scope. A dedicated meeting for a walk through is not mandatory. The reviews can be performed as peer reviews. The owner or responsible of the object to be reviewed has to distribute the review object to appropriate peer reviewers. It is mandatory to have at least two peer reviewers. Further the owner of the object has to set the date of the review, i.e. when he expects the review results to be back. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** none
Preparation by the reviewers	The peer reviewers will carefully study the review object. The main focus is to answer the related review checklist questions. If a question is not to be ticked as o.k. a comment has to be filled-in in the comment sheet. Further the peer reviewer has to apply his experience and common sense to detect problems and errors in the reviewed object which may not be covered by the review checklist questions. These have also to be reported in the comments. The prepared checklists and comment sheets have to be ready for the review meeting or alternatively have to be handed in to the responsible of the technical review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review.

Action	Action Description
	Participants: SW-D, SW-T, SW-PM depending on the scope of the review.
Perform the technical review meeting	The responsible of the technical review has to call a review meeting where the peer reviewers participate to reconcile and discuss the review findings. Alternatively he can do this without any participants, using the pre-filled comments and checklists of the peer reviewers. The responsible for the review has to set a status of the review i.e. "passed" (Finished) or "failed" (Not finished). The peer reviewers have to agree to this report by their signature. If the review status was "passed" (Finished) and the need for rework was not identified there are no further actions required. If the modifications to the object are only minor, an agreement has to be reached among the reviewers and the responsible, about what should be modified and the owner or responsible of the object has to rework without new review. If the status of the technical review was "failed" (Not finished), the responsible of the technical review has to modify the review object according to the findings of the review. This may involve discussions and agreements with the peer reviewers until a satisfactory solution is found and implemented. The goal is that the peer reviewers can agree to the modification. This means that the technical review has to be performed again on the modified review object and the described procedure has to be followed until the status can be set to "passed" (Finished). **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T depending on the scope of the review.
Documentation and tracking	The review results and related forms and documents. I.e. the review question list, comment list and the status result, have to be placed into the project CM archives. The labeling of theses files has to be performed to relate them to the appropriate baseline. All review findings which should lead to a modification of the review object have to be submitted to the problem resolution process. I.e. for the findings appropriate entries have to be made in a tracker database or the Action Item List of the project. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** none

2.3.2.13 Coding

Goal:	**Perform a source code implementation which complies to the SW source code implementation rules.** The source code which came into existence during the early design

	and simulation activities most likely will not be suitable for the micro-controller and other required platforms. The available source code has to be re-designed according to certain rules and defined structures.
Input:	Valid Software Design Document Source code of earlier design activities or prototypes
Output:	Software Source code complying to the implementation rules
Methods and Templates:	Software Design Method C Programming Guideline and Naming Conventions Source Code Unit and Header Templates

Action	**Action Description**
Perform source code implementation and refactoring	Use the Software Design Document and implement the source code according to the specifications and descriptions in this document. Make sure that you use the defined templates for headers and code files. If the source code was generated in various development cycles a refactoring may be necessary to bring it back to the defined standards. For the refactoring of source code you have to analyze the existing source code and set up test cases to be able to verify the functionality of the software after changes in the source code. Check the source code according to the defined rules and methods as described in: Software Design Method C Programming Guideline and Naming Conventions. Implement the refactoring results and check the refactored source code to make sure that the functionality is the same as in the simulation results. Note, that the coding and code refactoring is performed in various iterations during the design phase and with different scope for the various sample levels. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-D, SW-T

2.3.2.14 Perform a Technical Review on the Software Source Code

Goal:	**Establish a reviewed version of the work product** The goal is to review the completed version of a work product and to evaluate its suitability for the intended use.
Input:	Completed version of the work product to be reviewed
Output:	Reviewed version of the work product Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	Action Description
Plan and initiate the technical review	This description is valid for all technical reviews in the software development. The technical review (document review) has to be performed on the work product in scope. A dedicated meeting for a walk through is not mandatory. The reviews can be performed as peer reviews. The owner or responsible of the object to be reviewed has to distribute the review object to appropriate peer reviewers. It is mandatory to have at least two peer reviewers. Further the owner of the object has to set the date of the review, i.e. when he expects the review results to be back. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** none
Preparation by the reviewers	The peer reviewers will carefully study the review object. The main focus is to answer the related review checklist questions. If a question is not to be ticked as o.k. a comment has to be filled-in in the comment sheet. Further the peer reviewer has to apply his experience and common sense to detect problems and errors in the reviewed object which may not be covered by the review checklist questions. These have also to be reported in the comments. The prepared checklists and comment sheets have to be ready for the review meeting or alternatively have to be handed in to the responsible of the technical review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T, SW-PM depending on the scope of the review.
Perform the technical review meeting	The responsible of the technical review has to call a review meeting where the peer reviewers participate to reconcile and discuss the review findings. Alternatively he can do this without any participants, using the pre-filled comments and checklists of the peer reviewers. The responsible for the review has to set a status of the review i.e. "passed" (Finished) or "failed" (Not finished). The peer reviewers have to agree to this report by their signature. If the review status was "passed" (Finished) and the need for rework was not identified there are no further actions required. If the modifications to the object are only minor, an agreement has to be reached among the reviewers and the responsible, about what should be modified and the owner or responsible of the object has to rework without new review. If the status of the technical review was "failed" (Not finished), the responsible of the technical review has to modify the review object according to the findings of the review. This may involve discussions and agreements with the peer reviewers until a satisfactory solution is found and implemented. The goal is that the peer reviewers can agree to the modification. This means that the

Action	Action Description
	technical review has to be performed again on the modified review object and the described procedure has to be followed until the status can be set to "passed" (Finished). **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T depending on the scope of the review.
Documentation and tracking	The review results and related forms and documents. I.e. the review question list, comment list and the status result, have to be placed into the project CM archives. The labeling of theses files has to be performed to relate them to the appropriate baseline. All review findings which should lead to a modification of the review object have to be submitted to the problem resolution process. I.e. for the findings appropriate entries have to be made in a tracker database or the Action Item List of the project. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** none

2.3.2.15 Specify the Software Unit Tests

Goal:	**Specify the Software Unit Tests** A clear specification has to be made about all tests which will be employed. The document has to give detailed information to enable the tester to setup the environment for the tests and execute them test case by test case. The expected results and the Test End Criteria have to be clearly stated.
Input:	The valid Software Test Plan, the Software Design Document, containing the detailed design of the units to be tested.
Output:	SoftwareUnitTestSpecification.doc
Methods and Templates:	Software Testing Method SoftwareUnitTestSpecificationTemplate.doc

Action	Action Description
Define the Static Tests and document them in the Software Unit Test Specification	Define the relevant test cases to achieve the required static test coverage for: • The standard automatic code checks. These may be adapted to specific items as e.g. the CPU or compiler used in the project to be tested. • The MISRA rule set of the automatic code checks. The recommended part of the rules may be tailored for the project to be tested. • The code inspections. These may be adapted to specific items as e.g. the CPU or compiler used in the project to be

	tested. Document the test cases in the Software Unit Test Specification. Define the End of Test Criteria for the test cases and document them in the Software Unit Test Specification. Make sure that the related test objects are clearly identified. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-T, SW-TM
Define the Dynamic Tests and document them in the Software Unit Test Specification	Specify test cases for the dynamic unit test focusing on correctness of C-functions and groups of functions or components, as they are defined in the Software Design Document. The tests shall include a check for robustness against unexpected input values, memory access errors, or wrong input ranges, etc.. Further the C0/C1 code coverage tests shall be specified, aiming to achieve 100% coverage. Document the unit test cases in the Software Unit Test Specification. Define the End of Test Criteria for the test cases and document them in the Software Unit Test Specification. Make sure that the related test objects are clearly identified. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-T, SW-TM

2.3.2.16 Perform a Technical Review on the Software Unit Test Specification

Goal:	**Establish a reviewed version of the work product** The goal is to review the completed version of a work product and to evaluate its suitability for the intended use.
Input:	Completed version of the work product to be reviewed
Output:	Reviewed version of the work product Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	**Action Description**
Plan and initiate the technical review	This description is valid for all technical reviews in the software development. The technical review (document review) has to be performed on the work product in scope. A dedicated meeting for a walk through is not mandatory. The reviews can be performed as peer reviews. The owner or responsible of the object to be reviewed has to distribute the review object to appropriate peer reviewers. It is mandatory to have at least two peer reviewers. Further the owner of the object has to set the date of the review, i.e. when he expects the review results to be back. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs.

Action	Action Description
	Responsible: SW-PM **Participants:** none
Preparation by the reviewers	The peer reviewers will carefully study the review object. The main focus is to answer the related review checklist questions. If a question is not to be ticked as o.k. a comment has to be filled-in in the comment sheet. Further the peer reviewer has to apply his experience and common sense to detect problems and errors in the reviewed object which may not be covered by the review checklist questions. These have also to be reported in the comments. The prepared checklists and comment sheets have to be ready for the review meeting or alternatively have to be handed in to the responsible of the technical review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T, SW-PM depending on the scope of the review.
Perform the technical review meeting	The responsible of the technical review has to call a review meeting where the peer reviewers participate to reconcile and discuss the review findings. Alternatively he can do this without any participants, using the pre-filled comments and checklists of the peer reviewers. The responsible for the review has to set a status of the review i.e. "passed" (Finished) or "failed" (Not finished). The peer reviewers have to agree to this report by their signature. If the review status was "passed" (Finished) and the need for rework was not identified there are no further actions required. If the modifications to the object are only minor, an agreement has to be reached among the reviewers and the responsible, about what should be modified and the owner or responsible of the object has to rework without new review. If the status of the technical review was "failed" (Not finished), the responsible of the technical review has to modify the review object according to the findings of the review. This may involve discussions and agreements with the peer reviewers until a satisfactory solution is found and implemented. The goal is that the peer reviewers can agree to the modification. This means that the technical review has to be performed again on the modified review object and the described procedure has to be followed until the status can be set to "passed" (Finished). **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T depending on the scope of the review.
Documentation and tracking	The review results and related forms and documents. I.e. the review question list, comment list and the status result, have to be placed into the project CM archives. The labeling of theses files has to be performed to relate them to the appropriate baseline. All review findings which should lead to a modification of the review object

Action	Action Description
	have to be submitted to the problem resolution process. I.e. for the findings appropriate entries have to be made in a tracker database or the Action Item List of the project. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** none

2.3.2.17 Prepare and Perform the Software Unit Tests

Goal:	**Prepare and Perform the Software Unit Tests** The specific goals of this tasks include: • Approval that each unit is compliant to the relevant programming standards and rules as e.g. the MISRA rules and software discipline proprietary coding and design rules. • Approval that each unit is compliant to its underlying documentation. • Appropriate coverage of detailed design by test cases. • Check of the robustness of the code e.g. against boundary and overflow/underflow problems. • Check of the interfaces of C-functions and components. • Achieve the specified code coverage C0/C1. • Generation of a set of regression tests for the C-functions and components of the software. • Confidence that each unit is of ”good” quality Scope is the current Software version under development
Input:	The valid Software Test Plan, the valid Software Unit Test Specification, test objects (units)
Output:	SoftwareUnitTestReport.doc, verified test objects
Methods and Templates:	Software Testing Method SoftwareUnitTestReportTemplate.doc

Action	Action Description
Perform the Static Unit Tests	Install and configure the test and measurement tools which are used to support the static unit tests e.g. PC-Lint and other code analysis tools. Perform the static unit tests as e.g. PC-Lint, MISRA checks, and code inspection until the End of Test Criteria are reached as defined in the Software Unit Test Specification. Make sure that the related test objects are clearly identified and the task is properly commissioned to the test group by a filled in commissioning document. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-T **Participants:** none

Perform the Dynamic Unit Tests	Build up / install the test environment for dynamic tests, e.g. the Perl script environment or tools to measure the code coverage. Run the tests using the methods and input data until the defined End of Test Criteria are reached as defined in the Software Unit Test Specification. Perform the code coverage tests as specified. Record the output data for each test case (test log). Compare the output data with the expected output data. Make sure that the related test objects are clearly identified and the task is properly commissioned to the test group by a filled in commissioning document. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-T **Participants:** none
Generate a test report	Write the Software Unit Test Report and fill in the details about the found anomalies. In case of detected errors, enter the relevant data into the defect database. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-T **Participants:** none

2.3.2.18 Perform a S2 Management Review

Goal:	**Generate an agreement to enter the next development phase** The goal is to evaluate the completed work products of the previous development phase and to determine their suitability for the next development steps. This description is valid for all management reviews in the software development.
Input:	Completed versions of the work products of the previous development phase
Output:	Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	Action Description
Plan and initiate the management review	The management review has to be performed on the project progress of the project in scope. The SW-PM has to distribute the documents of the previous development phase to the review participants and allow enough time for preparation between the distribution and the review meeting. The SW-PM has to set the date of the review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** none
Perform the management	The SW-PM has to call a review meeting where the project progress will be evaluated. The focus in the meeting is to confirm that the

Action	Action Description
review meeting	work products of the previous development phase are present, technically reviewed and of sufficient quality to continue with the next step in the software development process. Further, the scope is to check on the project performance concerning the schedule, resources and quality. The participants of the review have to set a status of the review i.e. "passed" or "failed". The participants have to agree to this report by their signature. If the review status was "passed" there is the possibility that further actions are required but without a need of a new review. If the required activities in the project are only minor, an agreement has to be reached among the reviewers, about required activities and their tracking. If the status of the management review was "failed", the SW-PM has to solve the found problems and call for a repeated review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-Management, SW-D, SW-T depending on the scope of the review.

2.3.2.19 Milestone S2

This milestone marks the end of the design and implementation phase for the software project and the start of the verification and validation phase. The successful passing of the S2 management review constitutes automatically the reaching of the S2 milestone.

2.3.2.20 Specify the Software Integration Tests

Goal:	**Specify the Software Integration Tests** A clear specification has to be made about all tests which will be employed. The document has to give detailed information to enable the tester to setup the environment for the tests and execute them test case by test case. The expected results and the Test End Criteria have to be clearly stated
Input:	The valid Software Test Plan, the valid Software Requirements Specification.
Output:	Valid document: SoftwareIntegrationTestSpecification.doc
Methods and Templates:	Software Testing Method SoftwareIntegrationTestSpecificationTemplate.doc

Action	Action Description
Generate the Software Integration and Validation Test	Define the relevant test cases to achieve the required test coverage to ensure that the software meets the requirements, i.e. regarding functionality, performance, internal and external software interfaces, non-functional requirements, stress/load issues. Define the End of

Specification	Test Criteria for the test cases and document them in the Software Integration Test Specification. Make sure that the related test objects are clearly identified. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-T, SW-TM

2.3.2.21 Perform a Technical Review on the Software Integration Test Specification

Goal:	**Establish a reviewed version of the work product** The goal is to review the completed version of a work product and to evaluate its suitability for the intended use.
Input:	Completed version of the work product to be reviewed
Output:	Reviewed version of the work product Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	**Action Description**
Plan and initiate the technical review	This description is valid for all technical reviews in the software development. The technical review (document review) has to be performed on the work product in scope. A dedicated meeting for a walk through is not mandatory. The reviews can be performed as peer reviews. The owner or responsible of the object to be reviewed has to distribute the review object to appropriate peer reviewers. It is mandatory to have at least two peer reviewers. Further the owner of the object has to set the date of the review, i.e. when he expects the review results to be back. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** none
Preparation by the reviewers	The peer reviewers will carefully study the review object. The main focus is to answer the related review checklist questions. If a question is not to be ticked as o.k. a comment has to be filled-in in the comment sheet. Further the peer reviewer has to apply his experience and common sense to detect problems and errors in the reviewed object which may not be covered by the review checklist questions. These have also to be reported in the comments. The prepared checklists and comment sheets have to be ready for the review meeting or alternatively have to be handed in to the responsible of the technical review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review.

Action	Action Description
	Participants: SW-D, SW-T, SW-PM depending on the scope of the review.
Perform the technical review meeting	The responsible of the technical review has to call a review meeting where the peer reviewers participate to reconcile and discuss the review findings. Alternatively he can do this without any participants, using the pre-filled comments and checklists of the peer reviewers. The responsible for the review has to set a status of the review i.e. "passed" (Finished) or "failed" (Not finished). The peer reviewers have to agree to this report by their signature. If the review status was "passed" (Finished) and the need for rework was not identified there are no further actions required. If the modifications to the object are only minor, an agreement has to be reached among the reviewers and the responsible, about what should be modified and the owner or responsible of the object has to rework without new review. If the status of the technical review was "failed" (Not finished), the responsible of the technical review has to modify the review object according to the findings of the review. This may involve discussions and agreements with the peer reviewers until a satisfactory solution is found and implemented. The goal is that the peer reviewers can agree to the modification. This means that the technical review has to be performed again on the modified review object and the described procedure has to be followed until the status can be set to "passed" (Finished). **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T depending on the scope of the review.
Documentation and tracking	The review results and related forms and documents. I.e. the review question list, comment list and the status result, have to be placed into the project CM archives. The labeling of theses files has to be performed to relate them to the appropriate baseline. All review findings which should lead to a modification of the review object have to be submitted to the problem resolution process. I.e. for the findings appropriate entries have to be made in a tracker database or the Action Item List of the project. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** none

2.3.2.22 Prepare and Perform the Software Integration Tests

Goal:	**Prepare and Perform the Software Integration Tests** The specific goals of this tasks include: • Appropriate coverage of the relevant requirements and

	architecture elements by test cases. - Approval that the software interacts correctly between its units and the micro controller hardware and that the resource consumption is as expected. - Generation of a set of regression tests for the functionality of the Software. Scope is the current software version under development
Input:	The valid Software Test Plan, the valid Software Integration Test Specification, test objects (software units)
Output:	Valid document: SoftwareIntegrationTestReport.doc, verified test objects, test logs
Methods and Templates:	Software Testing Method SoftwareIntegrationTestReportTemplate.doc

Action	**Action Description**
Perform the Software Integration and Validation Tests	Build up / install test environment for integration and functional tests. Run the tests using the methods and input data until the End of Test Criteria are reached as defined in the Software Integration Test Specification. Record the output data for all test cases (test log). Compare the output data with the expected output data. Make sure that the related test objects are clearly identified and the task is properly commissioned to the test group by a filled in commissioning document. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-T **Participants:** none
Generate a test report	Document the test results in the Software Integration Test Report and in case of detected errors; enter the relevant data in the defect database. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-T **Participants:** none

2.3.2.23 Problem Resolution Meeting

Goal:	**Plan the implementation of bug fixes found in testing** The goal is to discuss the test findings, evaluate their criticality and decide on a time frame for the bug fixes. This is especially necessary if the tests are performed by a separate test group.
Input:	Any of the test reports: Software Unit Test Report Software Integration and Validation Test Report
Output:	Entries in the problem resolution database or Action Item List of the project.
Methods and	ActionItemListTemplate.doc

Templates:	

Action	Action Description
Discuss and plan the bug fixes	The meeting has the aim to review the test report(s) and derive actions from it. The tester has to explain why he considers portions in the code as defects and he shall explain their criticallity. The project members have to decide which bugs will be fixed in which release. The findings from the test shall be carried over into the problem resolution database or the project's Action Item List. Bugs may be grouped together when they are entered in the database. It is recommended to make the grouping according to the possible and planned implementation of the bug fixes and the planned releases. In case of no findings in the test reports the problem resolution meeting can be omitted. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-D, SW-T

2.3.2.24 Specify the Software Validation Tests

Goal:	**Specify the Functional Software Tests** A clear specification has to be made about all tests which will be employed. The document has to give detailed information to enable the tester to setup the environment for the tests and execute them test case by test case. The expected results and the Test End Criteria have to be clearly stated
Input:	The valid Software Test Plan, the valid Software Requirements Specification.
Output:	Valid document: SoftwareValitationTestSpecification.doc
Methods and Templates:	Software Testing Method SoftwareValidationTestSpecificationTemplate.doc

Action	Action Description
Generate the Software Integration and Validation Test Specification	Define the relevant test cases to achieve the required test coverage to ensure that the software meets the requirements, i.e. regarding functionality, performance, external software interfaces, non-functional requirements, and stress/load issues. Define the End of Test Criteria for the test cases and document them in the Software Integration and Validation Test Specification. Make sure that the related test objects are clearly identified. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-T, SW-TM

2.3.2.25 Perform a Technical Review on the Software Validation Test Specification

Goal:	**Establish a reviewed version of the work product** The goal is to review the completed version of a work product and to evaluate its suitability for the intended use.
Input:	Completed version of the work product to be reviewed
Output:	Reviewed version of the work product Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	**Action Description**
Plan and initiate the technical review	This description is valid for all technical reviews in the software development. The technical review (document review) has to be performed on the work product in scope. A dedicated meeting for a walk through is not mandatory. The reviews can be performed as peer reviews. The owner or responsible of the object to be reviewed has to distribute the review object to appropriate peer reviewers. It is mandatory to have at least two peer reviewers. Further the owner of the object has to set the date of the review, i.e. when he expects the review results to be back. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** none
Preparation by the reviewers	The peer reviewers will carefully study the review object. The main focus is to answer the related review checklist questions. If a question is not to be ticked as o.k. a comment has to be filled-in in the comment sheet. Further the peer reviewer has to apply his experience and common sense to detect problems and errors in the reviewed object which may not be covered by the review checklist questions. These have also to be reported in the comments. The prepared checklists and comment sheets have to be ready for the review meeting or alternatively have to be handed in to the responsible of the technical review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T, SW-PM depending on the scope of the review.
Perform the technical review meeting	The responsible of the technical review has to call a review meeting where the peer reviewers participate to reconcile and discuss the review findings. Alternatively he can do this without any participants, using the pre-filled comments and checklists of the peer reviewers. The responsible for the review has to set a status of the review i.e. "passed" (Finished) or "failed" (Not finished). The peer reviewers have to agree to this report by their signature. If the review status was "passed" (Finished) and the need for rework was not identified

Action	Action Description
	there are no further actions required. If the modifications to the object are only minor, an agreement has to be reached among the reviewers and the responsible, about what should be modified and the owner or responsible of the object has to rework without new review. If the status of the technical review was "failed" (Not finished), the responsible of the technical review has to modify the review object according to the findings of the review. This may involve discussions and agreements with the peer reviewers until a satisfactory solution is found and implemented. The goal is that the peer reviewers can agree to the modification. This means that the technical review has to be performed again on the modified review object and the described procedure has to be followed until the status can be set to "passed" (Finished). **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** SW-D, SW-T depending on the scope of the review.
Documentation and tracking	The review results and related forms and documents. I.e. the review question list, comment list and the status result, have to be placed into the project CM archives. The labeling of these files has to be performed to relate them to the appropriate baseline. All review findings which should lead to a modification of the review object have to be submitted to the problem resolution process. I.e. for the findings appropriate entries have to be made in a tracker database or the Action Item List of the project. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-D, SW-T, SW-PM depending on the scope of the review. **Participants:** none

2.3.2.26 Prepare and Perform the Software Validation Tests

Goal:	**Prepare and Perform the Functional Software Tests** The specific goals of this tasks include: • Appropriate coverage of the requirements (especially customer requirements) by test cases. • Approval that the software is compliant to the requirements. • Generation of a set of regression tests for the functionality of the Software. Scope is the current software version under development
Input:	The valid Software Test Plan, the valid Software Validation Test Specification, test objects (software units)
Output:	Valid document: SoftwareValidationTestReport.doc, verified test objects, test logs

Methods and Templates:	Software Testing Method SoftwareValidationTestReportTemplate.doc

Action	Action Description
Perform the Software Integration and Validation Tests	Build up / install test environment for integration and functional tests. Run the tests using the methods and input data until the End of Test Criteria are reached as defined in the Software Validation Test Specification. Record the output data for all test cases (test log). Compare the output data with the expected output data. Make sure that the related test objects are clearly identified and the task is properly commissioned to the test group by a filled in commissioning document. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-T **Participants:** none
Generate a test report	Document the test results in the Software Validation Test Report and in case of detected errors; enter the relevant data in the defect database. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-T **Participants:** none

2.3.2.27 Problem Resolution Meeting

Goal:	**Plan the implementation of bug fixes found in testing** The goal is to discuss the test findings, evaluate their criticality and decide on a time frame for the bug fixes. This is especially necessary if the tests are performed by a separate test group.
Input:	Any of the test reports: Software Unit Test Report Software Integration and Validation Test Report
Output:	Entries in the problem resolution database or Action Item List of the project.
Methods and Templates:	ActionItemListTemplate.doc

Action	Action Description
Discuss and plan the bug fixes	The meeting has the aim to review the test report(s) and derive actions from it. The tester has to explain why he considers portions in the code as defects and he shall explain their criticality. The project members have to decide which bugs will be fixed in which release. The findings from the test shall be carried over into the problem resolution database or the project's Action Item List. Bugs may be grouped together when they are entered in the database. It is recommended to make the grouping according to the possible and planned implementation of the bug fixes and the planned releases.

Action	Action Description
	In case of no findings in the test reports the problem resolution meeting can be omitted. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-D, SW-T

2.3.2.28 Perform a S3 Management Review

Goal:	**Generate an agreement to proceed with the software release** The goal is to evaluate the completed work products of the previous development phase and to determine their suitability for the software release.
Input:	Completed versions of the work products of the previous development phase
Output:	Review report and comment list: SoftwareReviewChecklist.doc
Methods and Templates:	Software Review Method SoftwareReviewChecklistTemplate.doc

Action	Action Description
Plan and initiate the management review	The management review has to be performed on the project progress of the project in scope. The SW-PM has to distribute the documents of the previous development phase to the review participants and allow enough time for preparation between the distribution and the review meeting. The SW-PM has to set the date of the review. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** none
Perform the management review meeting	The SW-PM has to call a review meeting where the project progress will be evaluated. The focus in the meeting is to confirm that the work products of the previous development phase are present, technically reviewed and of sufficient quality to continue with the final software release. Further, the scope is to check on the project performance concerning the schedule, resources and quality. The participants of the review have to set a status of the review i.e. "passed" or "failed". The participants have to agree to this report by their signature. If the review status was "passed" there is the possibility that further actions are required but without a need of a new review. If the required activities in the project are only minor, an agreement has to be reached among the reviewers, about required activities and their tracking. If the status of the management review was "failed", the SW-PM has to solve the found problems and call for a repeated review. **Possible Tailoring:** describe here in which cases and how the

Action	Action Description
	process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-Management, SW-D, SW-T depending on the scope of the review.

2.3.2.29 Milestone S3

This milestone marks the end of the verification and validation phase for the software project. The successful passing of the S3 management review constitutes automatically the reaching of the S3 milestone. The milestone also marks the end of the software development.

2.3.2.30 Release the Software

Goal:	**Officially release the software** By this step the software is officially released for use by the customer.
Input:	Milestone review report for the S3 review, with the status "passed"
Output:	Filled in SoftwareReleaseForm.doc
Methods and Templates:	SoftwareReleaseFormTemplate.doc

Action	Action Description
Release the Software	Check the management review report for the S3 review if it has the status "passed". Fill in the Software Release Form and distribute it according to the defined distribution list. **Possible Tailoring:** describe here in which cases and how the process can be tailored according to project needs. **Responsible:** SW-PM **Participants:** SW-QE

2.3.3 Role Descriptions

2.3.3.1 The Software Project Manager

Role	Software Project Manager (SW-PM)
Reports to	The SW-PM reports to the Project Manager of a customer project for all technical issues related to the specific customer project. The SW-PM reports to the SW management for technical issues related to a generic project.
Responsibilities and purpose	The SW-PM shall plan, conduct, track and control the generic SW project or the SW subproject within a development project. In case of a generic Software project the SW-PM is responsible for the successful conduct of the SW project in terms of time, efforts and quality. In case of a development project for a customer the SW-PM is the main interface to the overall Project Manager and is responsible for the successful conduct of the SW subproject in terms of time, efforts and quality.
Preconditions (Inputs)	The SW-PM shall base the schedule of his activities on the Master Project Plan of the customer project which shows clearly defined milestones and customer deliveries. In case of a generic SW project there is no Master Project Plan and the schedule shall be based on the defined milestones and deliveries as agreed with the management at project set up. The SW-PM shall base his planning on a resource assignment which was clearly agreed with the SW management. The SW-PM shall be clearly informed about the software development process which shall be applied for the project. The SW-PM shall be clearly informed about the software reuse and reusability goals which shall be applied for the project. The SW-PM shall be clearly informed about the software maintainability goals which shall be applied for the project. The SW-PM shall be clearly informed about other goals (as e.g. support of discipline related activities) which shall be accomplished by him.
Deliverables (Outputs)	The SW-PM shall generate and maintain the project specific Software Development Plan. The SW-PM shall coordinate and be responsible for all SW related activities in a project.

	The SW-PM shall be responsible to deliver the required SW source codes and intermediate work products (e.g. documents) in the required quality as defined in the software development process.
Quality and Process tasks	The SW-PM shall support and implement software development process improvement measures as defined by the software management. The SW-PM shall follow the instructions defined in the company wide as well as software related guidelines and procedures. The SW-PM shall follow the instructions defined in the Software Procedures, Software Quality Assurance Plan, Software Configuration Management Plan and all other binding guidelines as defined for the project. The SW-PM shall closely cooperate with the SW-QE to define and apply the project quality assurance measures in line with the applicable guidelines.
Standard tasks	The SW-PM shall perform the team building and staffing for the SW project. The SW-PM shall perform the effort estimation and time scheduling for the SW project. In case the SW project is a sub-project of a customer project this has to be coordinated with the Project Manager. The SW-PM shall perform the tracking of the SW project's progress and perform the project controlling and reporting. In case of a customer project the SW-PM shall contribute to the customer project's risk management. The SW-PM has the rights to access sufficient and adequate resources to achieve his defined goals, the possibility to escalate risks where necessary and to perform an independent and realistic reporting.
Skills and knowledge required	The SW-PM has to have detailed project management skills for software development projects. The SW-PM has to have leadership and communication skills to be able to coordinate activities and focus on the customer's satisfaction. The SW-PM has to have deep understanding and knowledge of the SW development process.

	The SW-PM has to have knowledge about the technical details of the product for which the software is generated. The SW-PM has to have detailed software discipline understanding.
Remarks	

2.3.3.2 The Software Configuration Manager

Role	Software Configuration Manager (SW-CM)
Reports to	The SW-CM reports to the Software Project Manager for all technical issues.
Responsibilities and purpose	The SW-CM shall be responsible to set up and maintain the archive spaces used by the project. He shall also perform or assist baselining and release activities for his assigned project.
Preconditions (Inputs)	The SW-CM shall base the schedule of his activities on the project specific plan. The SW-CM shall be clearly informed about the software development process which shall be applied for the project(s). The SW-CM shall be clearly informed about the technical goals which shall be applied for the project(s) and about the goal of CM in the software discipline. The SW-CM shall be clearly informed about other goals (as e.g. support of discipline related activities) which shall be accomplished by him.
Deliverables (Outputs)	The SW-CM shall set up the structures and documents in the project archive space. The SW-CM shall maintain the archive space to comply to the CM method of the discipline..
Quality and Process tasks	The SW-CM shall support and implement software process improvement measures as defined by the software discipline management. The SW-CM shall follow the instructions defined in the company wide as well as software related guidelines and procedures. The SW-CM shall follow the instructions defined in the Software Development Plan, Software Quality Assurance Plan, Software Configuration Management Plan and all other binding guidelines

	as defined for the project(s).
Standard tasks	The SW-CM shall set up the archive structures and initial files for the project according to the definitions in the related CM method. The SW-CM shall set up the tracking database for the project. The SW-CM shall generate the configuration management plan for the project. The SW-CM shall perform frequent informal CM audits on the project's archive space as well as tracking database. The SW-CM shall initiate or perform the corrective maintenance activities in case of any deviations from the CM method. This can be either on request by project members or triggered by CM audit results. The SW-CM shall support or perform baselining and support releases in the project. The SW-CM shall support or perform CM tool related activities, such as assigning rights or the installation of tools for project members.
Skills and knowledge required	The SW-CM has to have deep understanding and knowledge of configuration management principles. The SW-CM has to have detailed skills in the use of the configuration management tools applied in the project. The SW-CM has to have knowledge about the technical details of the product for which the configuration management is applied. The SW-CM has to have detailed software discipline understanding.
Remarks	

2.3.3.3 The Software Quality Engineer

Role	Software Quality Engineer (SW-QE)
Reports to	The SW-QE reports to the Project Quality Manager of a customer project for all technical issues. The disciplinary superior of the SW-QE must be separate from the Software domain, to guarantee independence.

Responsibilities and purpose	The SW-QE shall participate in the planning of the quality measures for the SW projects. The SW-QE shall track the status of the SW projects according to the defined quality plans of the projects. The SW-QE shall constitute an independent instance to check, evaluate and report the status of the SW projects.
Preconditions (Inputs)	The SW-QE shall base the schedule of his activities on the project plan of the SW project and perform the activities which are required according to the project progress. The SW-QE shall be clearly informed about the software development process which shall be applied for the project. The SW-QE shall be clearly informed about the software quality goals which shall be applied for the project.
Deliverables (Outputs)	The SW-QE shall generate and maintain the project specific Software Status Sheet and Software Quality Report. The SW-QE shall give his inputs and feedbacks in the review meetings for the work products and project milestones. The SW-QE has to approve by his signature that the required goals are met at the reviews, releases and other occasions where it is indicated by the SW procedures. The SW-QE shall intervene in situations where the required quality measures are not applied in an SW project and the risk of deliveries with insufficient quality is given. If necessary he has to escalate the subject.
Quality and Process tasks	The SW-QE shall closely cooperate with the SW process manager and support software process improvement measures as defined by the software management. The SW-QE shall follow the instructions defined in the company wide guidelines and procedures. The SW-QE shall follow the instructions defined in the Software Procedures, Software Quality Assurance Plan and all other binding guidelines as defined for the project. The SW-QE shall closely cooperate with the SW-PM to define and apply the project quality assurance measures in line with the applicable guidelines.
Standard tasks	The SW-QE shall check and evaluate the project status on a regular basis. The SW-QE shall generate and maintain the SW Status Sheet

	and SW Quality Report for the project and perform the defined reporting to the quality organization and management. The SW-QE shall participate in review meetings and status meetings of the projects as defined in the SW procedures. The SW-QE shall give a feedback about the quality status of the project to the SW-PM and if necessary define measures to improve the project quality in cooperation with the SW-PM. The SW-QE has to participate in certain activities as defined in the SW procedures (e.g. reviews and releases) and confirm by his signature that the expected quality is delivered.
Skills and knowledge required	The SW-QE has to have deep understanding and knowledge of the SW development process. The SW-QE has to have good communication skills to be able to solve conflicts, perform coaching and coordinate activities. The SW-QE has to have deep understanding and knowledge about the quality related international standards and methods. The SW-QE has to have a good software discipline understanding.
Remarks	

2.3.3.4 The Software Requirements Engineer

Role	Software Requirements Engineer (SW-RE)
Reports to	The SW-RE reports to the SW-PM for all technical issues.
Responsibilities and purpose	The SW-RE shall perform the requirements engineering for the SW project.
Preconditions (Inputs)	The SW-RE shall base the schedule of his activities on the project plan of the SW project and perform the related requirements engineering activities. The SW-RE shall be clearly informed about the software development process which shall be applied for the project. The SW-RE shall be clearly informed about the software project and software discipline goals.
Deliverables (Outputs)	The SW-RE shall generate and maintain the project specific Software Requirements Specification. The SW-RE shall participate in the review meetings which are related to the requirements engineering phase of the project.

Quality and Process tasks	The SW-RE shall follow the instructions defined in the company wide guidelines and procedures. The SW-RE shall follow the instructions defined in the Software Procedures, Software Requirements Engineering methods and guidelines and all other binding guidelines as defined for the project.
Standard tasks	The SW-RE shall collect all available requirements for the project as e.g. customer requirements, quality and legal requirements etc. The SW-RE shall collected requirements and analyse them. The goal is to improve their overall quality and make them complete and unambiguous. The SW-RE shall generate and maintain the software specific requirements specification for the project. The SW-RE shall participate in review meetings related to the requirements engineering phase of the project as defined in the SW procedures.
Skills and knowledge required	The SW-RE has to have detailed understanding and knowledge of the SW development process. The SW-RE has to have deep understanding and knowledge about requirements engineering methods and tools. The SW-RE has to have detailed software discipline understanding.
Remarks	

2.3.3.5 The Software Developer

Role	Software Developer (SW-D)
Reports to	The SW-D reports to the Software Project Manager for all technical issues.
Responsibilities and purpose	The SW-D shall be responsible to develop new software or perform the necessary changes to existing software to make it suitable for use in a customer project.
Preconditions (Inputs)	The SW-D shall base the schedule of his activities on the project specific plan. The SW-D shall be clearly informed about the software development process which shall be applied for the project(s). The SW-D shall be clearly informed about the technical goals which shall be applied for the project(s).

	The SW-D shall be clearly informed about other goals (as e.g. support of discipline related activities) which shall be accomplished by him.
Deliverables (Outputs)	The SW-D shall generate the source code for a software according to the technical needs or customer requirements. The SW-D shall generate the documentation related to the software as defined in the development process. This is especially the design document.
Quality and Process tasks	The SW-D shall support and implement software process improvement measures as defined by the software discipline management. The SW-D shall follow the instructions defined in the company wide as well as software related guidelines and procedures. The SW-D shall follow the instructions defined in the Software Development Plan, Software Quality Assurance Plan, Software Configuration Management Plan and all other binding guidelines as defined for the project(s).
Standard tasks	The SW-D shall participate in the necessary requirements engineering for software. The SW-D shall perform the software design according to the related design standards to satisfy the requirement for the software in the best possible way. The SW-D shall establish the software design document to document this software design. The SW-D shall establish the source code according to the related design and coding guidelines. The SW-D is responsible to support or perform the necessary tests for a software prior to its use in a sample or production delivery.
Skills and knowledge required	The SW-D has to have deep understanding and knowledge of signal processing. The SW-D has to have detailed knowledge of vehicle restraint systems and the involved physical principles. The SW-D has to have detailed software design and programming skills. The SW-D has to have detailed understanding and skills to use the tools necessary to perform software development. The SW-D has to have knowledge about the technical details of the product for which the software is established.

	The SW-D has to have detailed software discipline understanding.
Remarks	

2.3.3.6 The Software Test Manager

Role	Software Test Manager (SW-TM)
Reports to	The SW-TM reports to the Software Project Leader for all technical issues.
Responsibilities and purpose	The SW-TM shall define the project specific Software test concept in detail. The SW-TM shall ensure that the test concept is applied throughout the project(s). The SW-TM shall ensure that all software products and deliveries are tested as specified.
Preconditions (Inputs)	The SW-TM shall base the schedule of his activities on the Master Software Project Plan which shows clearly defined milestones and customer deliveries. The SW-TM shall base his planning on a resource assignment which was clearly agreed with the Software Project Leader and/or Software Discipline Leader. The SW-TM shall be clearly informed about the software development process which shall be applied for the project(s). The SW-TM shall be clearly informed about the software reuse and reusability goals which shall be applied for the project(s). The SW-TM shall be clearly informed about the software maintainability goals which shall be applied for the project(s). The SW-TM shall be clearly informed about other goals (as e.g. support of discipline related activities) which shall be accomplished by him.
Deliverables (Outputs)	The SW-TM shall generate and maintain the project specific Software Test Plan. The SW-TM shall coordinate and be responsible for the generation of test case specifications and related test software/scripts for regression tests. The SW-TM shall issue test reports.
Quality and Process tasks	The SW-TM shall support and implement software process improvement measures as defined by the software discipline management.

	The SW-TM shall follow the instructions defined in the company wide as well as software related guidelines and procedures. The SW-TM shall follow the instructions defined in the Software Development Plan, Software Quality Assurance Plan, Software Configuration Management Plan and all other binding guidelines as defined for the project(s). The SW-TM shall closely cooperate with the SW-QE to define and apply the project test concept in line with the applicable guidelines.
Standard tasks	The SW-TM shall define a test concept for the project which includes static tests (inspections and automatic code checker), and dynamic tests (using a test environment to execute the software). The SW-TM shall define and check the software test coverage, "end of test" criteria as well as the acceptance test criteria for sub-contracted software for the project. The SW-TM shall define the test environment in which the tests are performed. The SW-TM shall coordinate the generation of test case specifications and tests (regression tests) which are in line with the product and process requirements. Especially these are unit tests, software integration and software validation tests. The SW-TM shall track the test specifications in order to assure that all software requirements are tested. The SW-TM shall take part in the execution of tests which follow strictly established test criteria and clearly defined test cases. The SW-TM shall support the project team in analysing the errors found/spotted by the customer. The SW-TM shall support the SW-PM in the generation of plans and schedules. The SW-TM shall cooperate with the SW-QE in establishing the test concepts in line with the Software Quality Assurance Plan. The SW-TM shall prepare, record and evaluate suitable test data fit for generating project specific performance data which can be reviewed and audited together with the SW-QE. The SW-TM shall report the project related test results to the project leader and SW-QE.

	The SW-TM shall track the project specific test results and the related bug fixes together with the Software Project Leader. The SW-TM shall evaluate and select test tools. The SW-TM shall define and support the configuration and application (e.g. code checker settings) of test tools.
Skills and knowledge required	The SW-TM has to have deep understanding and knowledge of test methods. The SW-TM has to have detailed programming skills for the programming language which is applied in the project. The SW-TM has to have knowledge about the technical details of the product for which the software is generated. The SW-TM has to have detailed software discipline understanding.
Remarks	The SW-TM usually has only technical leadership. Any leadership for employees is not required, but is not excluded. The role of the Software Project Leader and the role of the SW-TM **must not** be fulfilled by the same person. There has to be a separate instance for the testing.

2.3.3.7 The Software Tester

Role	Software Tester (SW-T)
Reports to	The SW-T reports to the SW-TM for all technical issues.
Responsibilities and purpose	The SW-T shall generate or participate in the generation of project specific test specifications and other test related documents. The SW-T shall execute the defined tests.
Preconditions (Inputs)	The SW-T shall base the schedule of his activities on the project specific test plans which are part of the overall planning for the test team. The SW-T shall be clearly informed about the software development process which shall be applied for the project(s). The SW-T shall be clearly informed about the software reuse and reusability goals which shall be applied for the project(s). The SW-T shall be clearly informed about the software maintainability goals which shall be applied for the project(s).

	The SW-T shall be clearly informed about other goals (as e.g. support of discipline related activities) which shall be accomplished by him.
Deliverables (Outputs)	The SW-T shall generate and maintain project specific test specifications and related test software/scripts for regression tests. The SW-T shall record the test results and support the SW-TM in the generation of test reports.
Quality and Process tasks	The SW-T shall support and implement software process improvement measures as defined by the software discipline management. The SW-T shall follow the instructions defined in the company wide as well as software related guidelines and procedures. The SW-T shall follow the instructions defined in the Software Development Plan, Software Quality Assurance Plan, Software Configuration Management Plan and all other binding guidelines as defined for the project(s).
Standard tasks	The SW-T shall support the definition of a test concept for the project which includes static tests (inspections and automatic code checker), and dynamic tests tests. The SW-T shall support the SW-TM and SW-PM in the generation of plans and schedules. The SW-T shall generate test specifications and tests (regression tests) which are in line with the product and process requirements. Especially these are unit tests, software integration tests and software validation tests. The SW-T shall set up the test environment in which the various tests are performed. The SW-T shall execute the defined tests which follow strictly established test criteria and clearly defined test cases. The SW-T shall record the project related test results in the defined manner. The SW-T shall support the evaluation and selection of test tools. The SW-T shall perform the configuration and application (e.g. code checker settings) of test tools. The SW-T shall support the project team in analysing the errors

	found/spotted by the customer.
Skills and knowledge required	The SW-T has to have deep understanding and knowledge of test methods. The SW-T has to have detailed programming skills for the programming language which is applied in the project. The SW-T has to have knowledge about the technical details of the product for which the software is generated. The SW-T has to have detailed software discipline understanding.
Remarks	In the workflow for the software tests the software tester takes a considerable part. This means that has to define and implement test cases, configure test tools and program test stubs. This requires a solid training about test methods and tools. Preferably the tester has to be dedicated to these test activities. It is not recommendable to scatter the test activities to normal software developers because the effectiveness of the tests is strongly related to the experience and dedication of the testers.