TECHNICAL REPORT

ISO/IEC

TR

29110-5-1-1

First edition
2012-09-15

Software engineering — Lifecycle profiles for Very Small Entities (VSEs) —

Part 5-1-1:
Management and engineering guide: Generic profile group: Entry profile

Ingénierie du logiciel — Profils de cycle de vie pour très petits organismes (TPO) —

Partie 5-1-1: Guide de gestion et d'ingénierie: Groupe de profil générique: Profil d'entrée

Contents

Page

Table of illustrations

Table of tables

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

In exceptional circumstances, when the joint technical committee has collected data of a different kind from that which is normally published as an International Standard ("state of the art", for example), it may decide to publish a Technical Report. A Technical Report is entirely informative in nature and shall be subject to review every five years in the same manner as an International Standard.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC TR 29110-5-1-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering.*

ISO/IEC 29110 consists of the following parts, under the general title *Software engineering — Lifecycle profiles for Very Small Entities (VSEs)*:

— *Part 1: Overview* [Technical Report]

— *Part 2: Framework and taxonomy*

— *Part 3: Assessment guide* [Technical Report]

— *Part 4: Profile specifications*

    — *Part 4-1: Generic profile group*

    — *Part 4-m: Profile group m*

— *Part 5: Management and engineering guide*

    — *Part 5-1-1: Generic profile group : Entry profile* [Technical Report]

    — *Part 5-1-2: Generic profile group : Basic profile* [Technical Report]

    — *Part 5-m-n: Management and engineering guide for profile group m profile n* [Technical Report]

Introduction

The software industry recognizes the value of Very Small Entities (VSEs) in contributing valuable products and services. For the purpose of ISO/IEC 29110, a Very Small Entity (VSE) is an entity (enterprise, organization, department or project) having up to 25 people. VSEs also develop and/or maintain software that is used in larger systems; therefore, recognition of VSEs as suppliers of high quality software is often required.

According to the Organization for Economic Co-operation and Development (OECD) SME and Entrepreneurship Outlook report (2005) 'SMEs constitute the dominant form of business organisation in all countries world-wide, accounting for over 95 % and up to 99 % of the business population depending on country'. The challenge facing OECD governments is to provide a business environment that supports the competitiveness of this large heterogeneous business population and that promotes a vibrant entrepreneurial culture.

From studies and surveys conducted, it is clear that the majority of International Standards do not address the needs of VSEs. Conformance with these standards is difficult, if not impossible. Subsequently VSEs have no, or very limited, ways to be recognized as entities that produce quality software in their domain. Therefore, VSEs are often cut off from some economic activities.

It has been found that VSEs find it difficult to relate International Standards to their business needs and to justify the application of the standards to their business practices. Most VSEs can neither afford the resources, in terms of number of employees, budget and time, nor do they see a net benefit in establishing software life cycle processes. To rectify some of these difficulties, a set of guides has been developed according to a set of VSE characteristics. The guides are based on subsets of appropriate standards elements, referred to as VSE Profiles. The purpose of a VSE profile is to define a subset of International Standards relevant to the VSE context, for example, processes and outcomes of ISO/IEC 12207 and products of ISO/IEC 15289.

ISO/IEC 29110, targeted by audience, has been developed to improve product and/or service quality, and process performance. See Table 1. ISO/IEC 29110 is not intended to preclude the use of different life cycles such as: waterfall, iterative, incremental, evolutionary or agile.

Table 1 — ISO/IEC 29110 target audience

ISO/IEC 29110	Title	Target audience
Part 1	Overview	VSEs, customers, assessors, standards producers, tool vendors, and methodology vendors.
Part 2	Framework and taxonomy	Standards producers, tool vendors and methodology vendors. Not intended for VSEs.
Part 3	Assessment guide	Assessors, customers, and VSEs
Part 4	Profile specifications	Standards producers, tool vendors and methodology vendors. Not intended for VSEs.
Part 5	Management and engineering guide	VSEs and customers

If a new profile is needed, ISO/IEC 29110-4 and ISO/IEC TR 29110-5 can be developed without impacting existing documents and they become ISO/IEC 29110-4-m and ISO/IEC 29110-5-m-n, respectively, through the ISO/IEC process.

ISO/IEC TR 29110-1 defines the business terms common to the VSE Profile Set of Documents. It introduces processes, lifecycle and standardization concepts, and the ISO/IEC 29110 series. It also introduces the characteristics and requirements of a VSE, and clarifies the rationale for VSE-specific profiles, documents, standards and guides.

ISO/IEC 29110 introduces the concepts for software engineering standardized profiles for VSEs, and defines the terms common to the VSE Profile Set of Documents. It establishes the logic behind the definition and application of standardized profiles. It specifies the elements common to all standardized profiles (structure, conformance, assessment) and introduces the taxonomy (catalogue) of ISO/IEC 29110 profiles.

ISO/IEC TR 29110-3 defines the process assessment guidelines and compliance requirements needed to meet the purpose of the defined VSEs Profiles. ISO/IEC TR 29110-3 also contains information that can be useful to developers of assessment methods and assessment tools. ISO/IEC TR 29110-3 is addressed to people who have direct relation with the assessment process, e.g. the assessor and the sponsor of the assessment, who need guidance on ensuring that the requirements for performing an assessment have been met.

ISO/IEC 29110-4-1 provides the specification for all the profiles of the Generic Profile Group. The Generic Profile Group is applicable to VSEs that do not develop critical software products. The profiles are based on subsets of appropriate standards elements. VSEs' Profiles apply and are targeted to authors/providers of guides and authors/providers of tools and other support material.

This part of ISO/IEC 29110 provides an implementation management and engineering guide for the Entry Profile of the Generic Profile Group described in ISO/IEC 29110-4-1. The Entry Profile describes software development of a single application by a single project team with no special risk or situational factors for start-up VSEs (i.e. VSEs who started their operation less than 3 years) and/or for VSEs working on small project (e.g. project size of less than 6 person-months).

Figure 1 describes the ISO/IEC 29110 series and positions the parts within the framework of reference. Overviews and guides are published as Technical Reports (TR), and profiles are published as International Standards (IS).

29110 Overview (TR 29110-1)

29110 Profiles (IS)

> **Framework and Taxonomy (IS 29110-2)**

> **Specifications of VSE Profiles (IS 29110-4)**
>> **Specification - VSE Profile Group m**
>> **(IS 29110-4-m)**

29110 Guides (TR)

> **Assessment Guide (TR 29110-3)**

> **Management and Engineering Guide (TR 29110-5)**
>> **Management and Engineering Guide VSE Profile m-n**
>> **(TR 29110-5-m-n)**
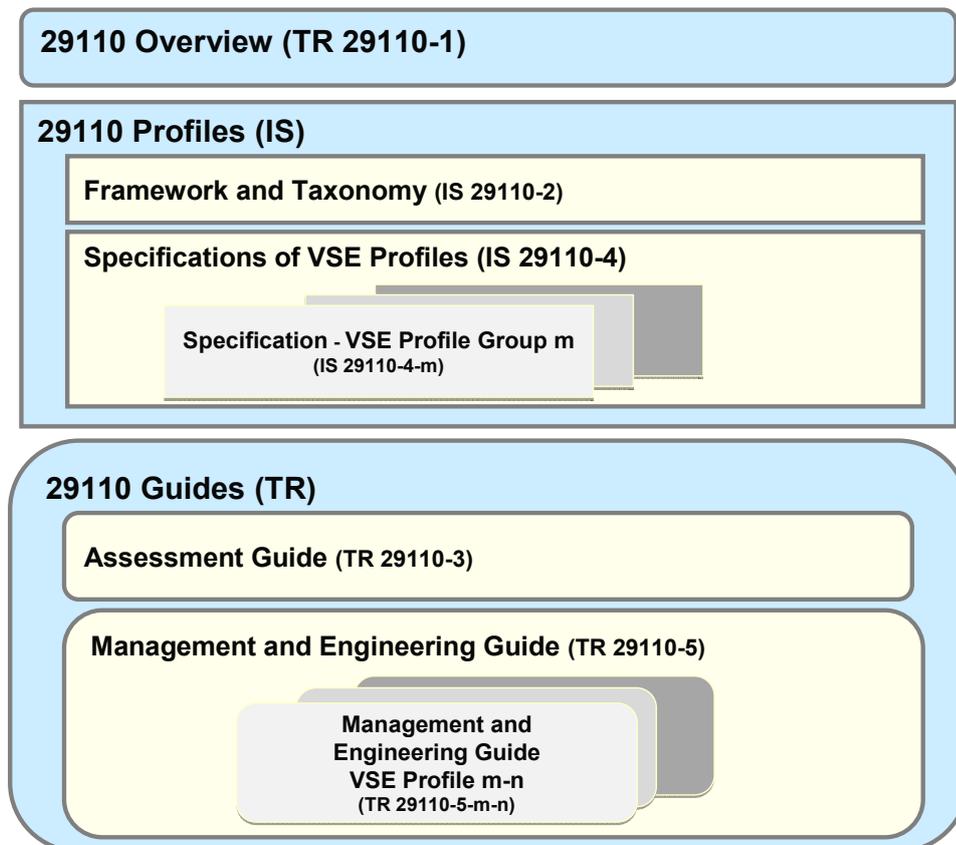
Figure 1 — ISO/IEC 29110 series

Software engineering — Lifecycle profiles for Very Small Entities (VSEs) —

Part 5-1-1:
Management and engineering guide: Generic profile group: Entry profile

1 Scope

1.1 Fields of application

This part of ISO/IEC 29110 is applicable to Very Small Entities (VSEs). VSEs are enterprises, organizations, departments or projects up to 25 people. The life cycle processes described in the set of International Standards (IS) Profiles and Technical Reports (TR) are not intended to preclude or discourage their use by organizations bigger than VSEs.

This part of ISO/IEC 29110 provides the management and engineering guide to the Entry profile described in ISO/IEC 29110-4-1 through project management and software implementation processes. This part of ISO/IEC 29110 is a stand alone guide. It is not intended for a VSE to use the standardized profile to implement this part of the ISO/IEC 29110.

This part of ISO/IEC 29110 applies for software development project. The project may be to fulfil an external or internal contract. The internal contract need not be explicit between the project team and their customer.

Using this part of ISO/IEC 29110, a VSE can obtain the following benefits:

— An agreed set of project requirements and expected products is delivered to the customer;

— A disciplined management process, that provides project visibility and corrective actions of project problems and deviations, is performed;

— A systematic software implementation process, that satisfies customer needs and ensures quality products, is followed.

1.2 Target Audience

This part of ISO/IEC 29110 is targeted at start-up VSEs (i.e. VSEs who started their operation less than 3 years) and/or at VSEs working on small project (e.g. project size of less than 6 person-months).

It is intended to be used with any processes, techniques and methods that enhance the VSE's Customer satisfaction and productivity.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC TR 29110-1, *Software Engineering — Lifecycle Profiles for Very Small Entities (VSEs) — Part 1: Overview*

3   Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC TR 29110-1 apply.

4   Symbols (and abbreviated terms)

4.1   Naming, diagramming and definition conventions

The following process structure description and notation are used to describe the processes:

Name – process identifier, followed by its abbreviation in brackets " ( )".

Purpose - general goals and results expected of the effective implementation of the process. The implementation of the process should provide tangible benefits to the stakeholders. The purpose is identified by the abbreviation of the process name.

Objectives - specific goals to ensure the accomplishment of the process purpose. The objectives are identified by the abbreviation of the process name, followed by the letter "O" and a consecutive number, for example PM.O1, SI.O2, etc. Each objective is followed by the square box which includes a list of the chosen processes for the entry profile from ISO/IEC 12207:2008 and its outcomes related to the objective.

Input Products – products required to perform the process and its corresponding source, which can be another process or an external entity to the project, such as the Customer. Identified by the abbreviation of the process name and showed as two column table of product names and sources.

Output Products – products generated by the process and its corresponding destination, which can be another process or an external entity to the project, such as Customer or Organizational Management. Identified by the abbreviation of the process name and showed as two column table of product names and destinations.

Internal Products – products generated and consumed by the process. Identified by the abbreviation of the process name and showed as one column table of the product names.

All products' names are printed in cursive and initiate with capital letters. Some products have one or more statuses attached to the product name surrounded by square brackets "[ ]"and separated by ",". The product status may change during the process execution. See Clause 6.5 and 7.5 for the alphabetical list of the products, its descriptions, possible statuses and the source of the product. The source can be another process or an external entity to the project, such as the Customer.

Roles involved – names and abbreviation of the functions to be performed by project team members. Several roles may be played by a single person and one role may be assumed by several persons. Roles are assigned to project participants based on the characteristics of the project. The role list is identified by the abbreviation of the process name and showed as two column table. See Clause 8 for the alphabetical list of the roles, its abbreviations and required competencies description.

Diagram – graphical representation of the processes. The large round-edged rectangles indicate process or activities and the smaller square-edged rectangles indicate the products. The directional or bidirectional thick arrows indicate the major flow of information between processes or activities. The thin directional or bidirectional arrows indicate the input or output products. The notation used in the diagrams does not imply the use of any specific process lifecycle.

Activity – a set of cohesive tasks. Task is a requirement, recommendation, or permissible action, intended to contribute to the achievement of one or more objectives of a process. A process activity is the first level of process workflow decomposition and the second one is a task. Activities are identified by process name abbreviation followed by consecutive number and the activity name.

Activity Description - each activity description is identified by the activity name and the list of related objectives surrounded by brackets "( )". For example PM.1 Project Planning (PM.O1, PM.O5, PM.O6, PM.O7) means that the activity PM.1 Project Planning contributes to the achievement of the listed objectives: PM.O1, PM.O5, PM.O6 and PM.O7. The activity description begins with the task summary and is followed by the task descriptions table. The task description doesn't impose any technique or method to perform it. The selection of the techniques or methods is left to the VSE or project team.

Tasks description table contain four columns corresponding to:

— Role- the abbreviation of roles involved in the task execution.

— Task - description of the task to be performed. Each task is identified by activity ID and consecutive number, for example PM1.1, PM1.2, and so on.

— Input Products - products needed to execute the task.

— Output Products - products created or modified by the execution of the task.

Incorporation to *Project Repository* – list of products to be saved in project repository; the *Version Control Strategy* has to be applied to some of them (see Clause 6.7.2 and 7.7.2). It is useful as a checklist for project manager and technical leader.

NOTE   Tables used in process description are for presentation purpose only.

Competent project team – People / Individuals performing work affecting quality of product or service need to be competent on the basis of appropriate education, training, skills and experience.

4.2   Abbreviations

The following abbreviations are used in this document:

VSE   Very Small Entity

5   Overview

The Entry VSE Profile Management and Engineering Guide applies to a Very Small Entity (VSE) (enterprise, organization, department or project up to 25 people) dedicated to software development. The project may fulfil an external or internal contract. The internal contract between the project team and its customer need not be explicit. The Entry profile is the first profile of the Generic profile group. The Entry profile is defined for the case when more flexible and more light-weight software process is needed than the Basic profile scope, e.g. for the case when user-risk is very low, using period is very short, and process responsibility is appropriately divided between the acquirer and the developer.

The Guide provides Project Management and Software Implementation processes which integrate practices based on the selection of ISO/IEC 12207:2008, *Systems and Software Engineering — Software Life Cycle Processes* and ISO/IEC 15289:2011, *Systems and software engineering — Content of life-cycle information products (documentation)* standards elements. Annex A provides information about Deployment Packages which will facilitate the implementation of these processes. Annex B provides information about a self-diagnostic.

This part of ISO/IEC 29110 is intended to be used by the VSE to establish processes to implement any development approach or methodology including, e.g., agile, evolutionary, incremental, test driven development, etc. based on the VSE organization or project needs.

Using the Guide, VSE can obtain benefits in the following aspects:

— An agreed set of project requirements and expected products is delivered to the customer;

— A systematic software implementation process, that satisfies customer needs and ensures quality products, is followed.

To use the Guide the VSE needs to fulfil the following entry conditions:

— Project statement of work is documented;

— Competent project team, including project manager, is assigned; and

— Goods and services to start the project are available.

The purpose of the Project Management process is to establish and carry out in a systematic way the tasks of the software implementation project, which allows complying with the project's objectives in the expected quality, time and cost.

The purpose of the Software Implementation process is the systematic performance of the analysis, software component identification, construction, integration and tests, and product delivery activities for new or modified software products according to the specified requirements.

Both processes are interrelated (see Figure 2).
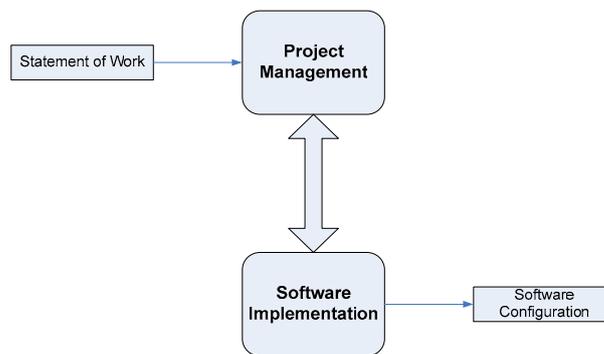
NOTE   Diagram notation is explained in 4.1.

Figure 2 — Entry profile guide processes

PM process uses the customer's statement of work to elaborate the project plan. The PM project assessment and control tasks compare the project progress against the project plan. The PM project closure activity delivers the software configuration, produced by SI, and gets the customer's acceptance to formalize the end of the project. A project repository is established to save the work products during the project.

The execution of the SI process is driven by the project plan. SI process starts with an initiation activity of the project plan revision. Project plan will guide the execution of the software requirements analysis, software component identification, software construction, software integration and test, and product delivery activities.

To remove product's defects verification, validation and test tasks are included in the activities workflow.

The customer provides a statement of work as an input to Project Management process and receives a software configuration as a result of Software Implementation process execution (see Figure 2).

6   Project Management (PM) process

6.1   PM purpose

The purpose of the Project Management process is to establish and carry out in a systematic way the tasks of the software implementation project, which allows complying with the project's objectives in the expected

quality, time and costs.

This part of ISO/IEC 29110 is intended to be used by the VSE to establish processes. The life cycle processes described are not intended to preclude the use of different life cycles such as: waterfall, iterative, incremental, evolutionary or agile.

6.2   PM objectives

PM.O1. The *Project Plan* for the execution of the project is developed according to *the Statement of Work* and reviewed and accepted by the Customer. The tasks and resources necessary to complete the work are sized and estimated.

> *6.3.1 Project Planning Process*
>
> *a)   the scope of the work for the project is defined;*
>
> *c)   the tasks and resources necessary to complete the work are sized and estimated;*
>
> *e)   plans for the execution of the project are developed; and*
>
> *f)   plans for the execution of the project are activated.*
>
>
> *6.3.7 Measurement Process*
>
> *a)   the information needs of technical and management processes are identified.*
>
> *[ISO/IEC 12207:2008, 6.3.1, 6.3.7]*

PM.O2. Progress of the project is monitored against the Project Plan and recorded in the Progress Status Record. Closure of the project is performed to get the Customer acceptance documented in the Acceptance Record.

6.3.2 Project Assessment and Control Process

a)   progress of the project is monitored and reported;

d)   project objectives are achieved and recorded.

6.3.7 Measurement Process

d)   the required data are collected, stored, analyzed, and the results interpreted; and

6.4.8   Software Acceptance Support Process

a)   the product is completed and delivered to the acquirer;

[ISO/IEC 12207:2008, 6.3.2, 6.3.7, 6.4.8]

PM.O3. The Changes Requests are addressed, evaluated and tracked.

7.1.2 Software Requirements Analysis Process

g)   changes to the software requirements are evaluated for cost, schedule and technical impact.

[ISO/IEC 12207:2008, 7.1.2]

PM.O4. Review meetings with the Work Team and the Customer are held. Agreements are registered and tracked.

7.2.6 Software Review Process

a)   management and technical reviews are held based on the needs of the project;

d)   action items resulting from reviews are tracked to closure.

[ISO/IEC 12207:2008, 7.2.6]

PM.O5. *Risks* are identified as they develop and during the conduct of the project.

6.3.4 Risk Management Process

c) *risks are identified as they develop and during the conduct of the project;*

7.2.6 Software Review Process

e) risks and problems are identified and recorded.

[ISO/IEC 12207:2008, 6.3.4, 7.2.6]

PM.O6. Items of Software Configuration are identified and controlled.

7.2.2 Software Configuration Management Process

b) *items generated by the process or project are identified, defined and baselined;*

g) *the storage, handling and delivery of the items are controlled.*

[ISO/IEC 12207:2008, 7.2.2]

PM.O7. Software Quality Assurance is performed to provide assurance that work products and processes comply with the *Project Plan* and *Requirements Specification*.

NOTE   The implementation of the Software Quality Assurance process is through the performance of the verifications, validations and review tasks performed in Project Management and Software Implementation processes.

7.2.3 Software Quality Assurance Process

b) *evidence of Software quality assurance is produced and maintained;*

d) *adherence of products, processes and activities to the applicable standards, procedures and requirements are verified.*

[ISO/IEC 12207:2008, 7.2.3]

6.3   PM input products

Table 2 — PM input products

Name	Source
Statement of Work	Customer
Software Configuration	Software Implementation
Change Request	Customer

6.4 PM output products

Table 3 — PM output products

Name	Destination
Project Plan	Software Implementation
Acceptance Record	Customer
Project Repository	Software Implementation
Meeting Record	Customer
Software Configuration	Customer

6.5 PM internal products

Table 4 — PM internal products

Name
Change Request
Meeting Record
Progress Status Record

6.6 PM roles involved

Table 5 — PM roles involved

Role	Abbreviation
Customer	CUS
Project Manager	PM
Work Team	WT

6.7 PM diagram

6.7.1 Overview

The following diagram shows the flow of information between the Project Management Process activities including the most relevant work products and their relationship.
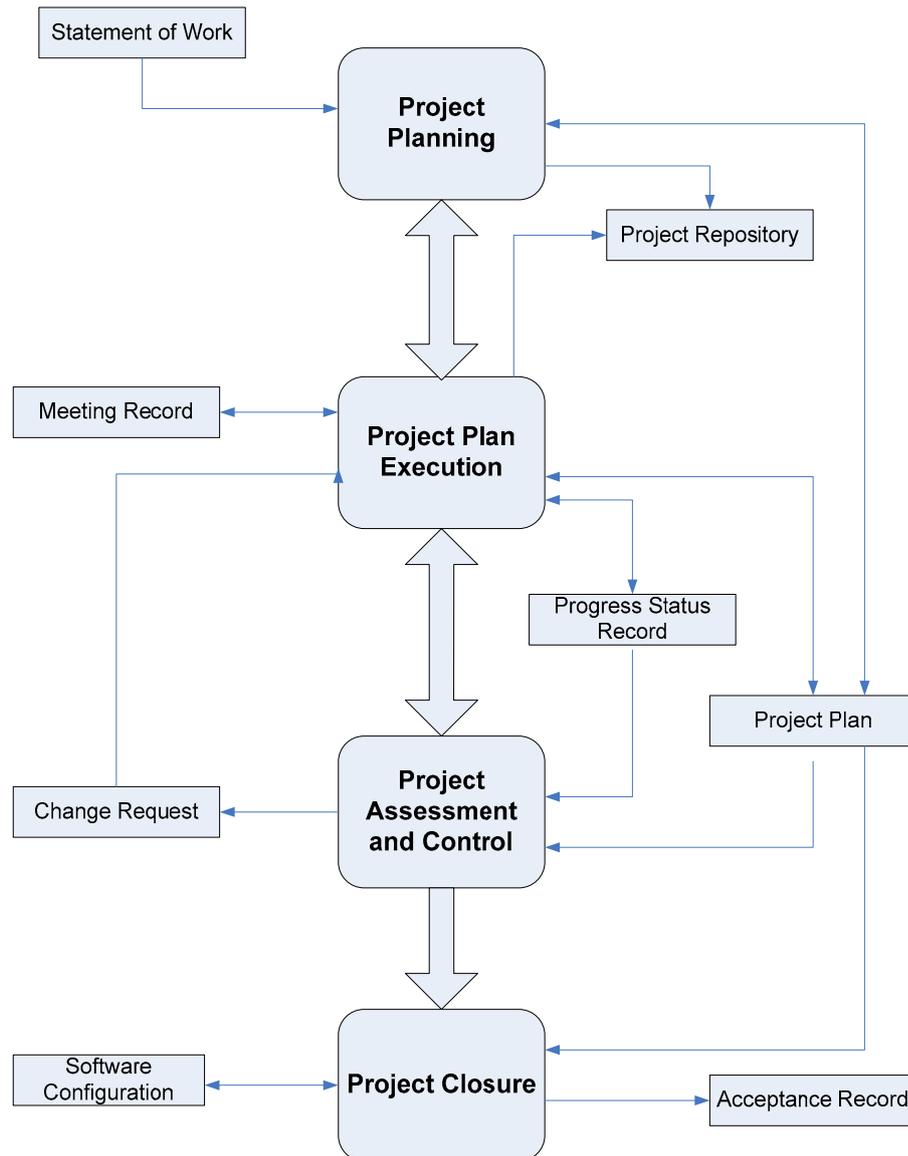
Figure 3 — Project Management process diagram

6.7.2 PM activities

The Project Management Process has the following activities:

— PM.1 Project Planning

— PM.2 Project Plan Execution

— PM.3 Project Assessment and Control

— PM.4 Project Closure

PM.1 Project planning, (PM.O1, PM.O5, PM.O6, PM.O7)

The Project Planning activity documents the planning details needed to manage the project. The activity provides:

— Reviewed *Statement of Work* and the tasks needed to provide the contract deliverables and to satisfy customer requirements.

— Project quality assurance approach through verification and validation of work products/deliverables, customer reviews.

— Work team and customer roles and responsibilities.

— Project resources needs.

— Estimates of effort, cost and schedule.

— Identified project risks.

— Project repository to store, handle and deliver controlled product and document versions and baselines.

Table 6 — PM.1 task list

Role	Task List	Input Products	Output Products
PM WT	PM.1.1 Review the *Statement of Work*	*Statement of Work*	*Statement of Work [reviewed]*
PM WT	PM.1.2 Identify the specific tasks to be performed in order to produce the deliverables and their software components identified in the *Statement of Work.* Include tasks in the SI process along with verification, validation and reviews with Customer and Work Team tasks to assure the quality of work products.	*Statement of Work[reviewed]*	*Project Plan* *- Tasks*
PM WT	PM.1.3 Establish the *Estimated Duration* to perform each task.	*Project Plan* *- Tasks*	*Project Plan* *- Estimated Duration*
PM WT	PM.1.4 Identify and document the resources: human, material, equipment and tools.	*Statement of Work*	*Project Plan* *- Resources*
PM WT	PM.1.5 Establish the Composition of Work Team assigning roles and responsibilities according to the Resources.	*Project Plan* *- Resources*	*Project Plan* *- Composition of Work Team*
PM WT	PM.1.6 Assign estimated start and completion dates to each one of the tasks in order to create the *Schedule of the Project Tasks.*	*Project Plan* *- Tasks* *- Estimated Duration* *- Composition of Work Team*	*Project Plan* *- Schedule of the Project Tasks*
PM	PM.1.7 Calculate and document the project *Estimated Effort and Cost.*	*Project Plan* *- Schedule of the Project Tasks* *- Resources*	*Project Plan* *- Estimated Effort and Cost*

Role	Task List	Input Products	Output Products
PM WT	PM.1.8 Identify and document the risks which may affect the project.	*All elements previously defined*	*Project Plan* *- Identification of Project Risks*
PM	PM.1.9 Generate the *Project Plan* integrating the elements previously identified and documented.	*All elements previously defined*	*Project Plan* *- Tasks* *- Estimated Duration* *- Resources* *- Composition of Work Team* *- Schedule of the Project Task* *- Estimated Effort and Cost* *- Identification of Project Risks*
PM CUS	PM.1.10 Review and accept appropriate parts of the *Project Plan.* Customer reviews and accepts the *Project Plan.*	*Project Plan*	*Project Plan [accepted]*
PM WT	PM.1.11 Establish the project repository.	*Project Plan*	*Project Repository*

PM.2 Project plan execution (PM.O2, PM.O3, PM.O4, PM.O5, PM.O7)

The Project Plan Execution activity implements the documented plan on the project. The activity provides:

— Monitoring the project against the *Project plan*.

— Status of the Project Plan Execution

— Change Request accepted by the Customer

— Reviews and agreements with the Customer

Table 7 — PM.2 task list

Role	Task List	Input Products	Output Products
PM WT	PM.2.1 Monitor and record status of the Project Plan execution.	*Project Plan*	*Progress Status Record*

Role	Task List	Input Products	Output Products
PM CUS WT	PM.2.2 Conduct meetings with the Customer, record agreements and track them to closure. Change Request initiated by Customer, needs to be negotiated to reach acceptance of both parties.	*Project Plan* *Progress Status Record* *Change Request*	*Meeting Record* *Change Request [accepted]*

PM.3 Project assessment and control (PM.O2)

The Project Assessment and Control activity evaluates the performance of the plan. The activity provides:

— Evaluation of actual plan performance and progress against targets.

— Track change requests.

— Documented problem, corrective action defined, and tacked to closure.

Table 8 — PM.3 task list

Role	Task List	Input Products	Output Products
PM WT	PM.3.1 Evaluate project progress with respect to the *Project Plan*, comparing: - actual tasks against planned tasks - actual resource allocation against planned resources - actual cost against budget estimates - actual time against planned schedule - actual risk against previously identified	*Project Plan* *Progress Status Record*	*Progress Status Record [evaluated]*
PM WT	PM.3.2 Evaluate and Track the changes request from customer.	*Change Request*	*Change Request [Tracked]*
PM WT	PM. 3.3 Establish actions to correct deviations or problems and track them to closure.	*Progress Status Record*	

PM.4 Project closure (PM.O2)

The Project Closure activity provides the project's documentation and products in accordance with *contract requirements*. The activity provides:

— Support of Customer product acceptance

— Completion of the project and sign of the Acceptance Record

— Summary and updated project repository for project closure

Table 9 — PM.4 task list

Role	Task List	Input	Output

PM CUS	PM.4.1. Formalize the completion of the project, providing acceptance support and getting the *Acceptance Record* signed.	*Project Plan* *Software Configuration [ready to be delivered]*	*Acceptance Record* *Software Configuration [accepted]*
PM	PM.4.2 Update *Project Repository*.	PM products - *Project Plan* - *Change Request* - *Progress Status - Record* - *Meeting Record* - *Acceptance Record* SI products - *Software Component Identification* - *Test Cases and Test Procedures* - *Software Components* - *Test Report* - *Software Configuration*	*Project Repository [updated]*

6.7.3   PM incorporation to Project Repository

The list of products to be saved in *Project Repository*.

Table 10 — PM repository products

Product
Project Plan
Change Request
Acceptance Record
Meeting Record

Progress Status Record

---

7   Software Implementation (SI) process

7.1   SI purpose

The purpose of the Software Implementation process is the systematic performance of the analysis, software component identification, construction, integration and tests, and product delivery activities for new or modified software products according to the specified requirements.

This part of ISO/IEC 29110 is intended to be used by the VSE to establish processes. The life cycle processes described are not intended to preclude the use of different life cycles such as: waterfall, iterative, incremental, evolutionary or agile.

7.2   SI objectives

SI.O1. Tasks of the activities are performed through the accomplishment of the current *Project Plan*.

SI.O2. Software requirements are defined, analyzed for correctness and testability, approved by the Customer, and communicated.

6.4.1 Stakeholder Requirements Definition Process

a)   the required characteristics and context of use of services are specified.

7.1.2 Software Requirements Analysis Process

a)   the requirements allocated to the software elements of the system and their interfaces are defined;

b)   software requirements are analyzed for correctness and testability;

f)   the software requirements are approved and updated as needed;

[ISO/IEC 12207:2008, 6.4.1, 7.1.2]

SI.O3. Software components and their interfaces are identified.

7.1.3 Software Architectural Design Process

a)   a software architectural design is developed and baselined that describes the software items that will implement the software requirements;

b)   internal and external interfaces of each software item are defined;

[ISO/IEC 12207:2008, 7.1.3]

SI.O4. Software components are produced. Unit test are performed to verify the consistency with software requirements.

> *7.1.5 Software Construction Process*
>
> *b)   software units defined by the design are produced;*
>
> *d)   verification of the software units against the requirements and the design is accomplished.*
>
> *[ISO/IEC 12207:2008, 7.1.5]*

SI.O5. Software is produced. Software components are integrated and verified using *Test Cases and Test Procedures*. Results are recorded at the *Test Report*. Defects are corrected.

> *7.1.6 Software Integration Process*
>
> *c)   software items are verified using the defined criteria;*
>
> *d)   software items defined by the integration strategy are produced;*
>
> *e)   results of integration testing are recorded;*
>
> *7.1.7 Software Qualification Testing Process*
>
> *a)   criteria for the integrated software is developed that demonstrates compliance with the software requirements;*
>
> *b)   integrated software is verified using the defined criteria; and*
>
> *c)   test results are recorded.*
>
> *[ISO/IEC 12207:2008, 7.1.6, 7.1.7]*

SI.O6. Software configuration is prepared for delivery.

> *6.1.2 Supply Process*
>
> *d)   a product and/or service that meets the agreed requirements are developed by the supplier;*
>
> *e)   the product and/or service is delivered to the acquirer in accordance with the agreed requirements; and*
>
> *f)   the product is installed in accordance with the agreed requirements.*
>
> *7.2.1 Software Documentation Management Process*
>
> *c)   documentation to be produced by the process or project is identified; and*
>
> *e)   documentation is developed and made available in accordance with identified standards.*
>
> *[ISO/IEC 12207:2008, 6.1.2, 7.2.1]*

SI.O7. Verification and Validation tasks of all required work products are performed to achieve consistency among output and input products in each activity. Defects are identified and corrected.

7.2.4 Software Verification Process

c)   required verification activities are performed;

7.2.5 Software Validation Process

c)   required validation activities are performed;

[ISO/IEC 12207:2008, 7.2.4, 7.2.5]

NOTE   It's not the intention that all verification activities and work products are made available to the customer. Verifications should be performed by individuals that have organizational freedom, authority, to permit objective evaluation, and to initiate, effect, resolve and verify problem resolution.

7.3   SI input products

Table 11 — SI input products

Name	Source
Project Plan	Project Management
Project Repository	Project Management

7.4   SI output products

Table 12 — SI output products

Name	Destination
Software Configuration • *Requirements Specification* • *Software*	Project Management

7.5   SI internal products

Table 13 — SI internal products

Name
Software Component Identification
Test Cases and Test Procedures
Software Component
Test Report

7.6 SI roles involved

Table 14 — SI roles involved

Role	Abbreviation
Customer	CUS
Project Manager	PM
Work Team	WT

7.7 SI diagram

The following diagram shows the flow of information between the Software Implementation Process activities including the most relevant work products and their relationship.
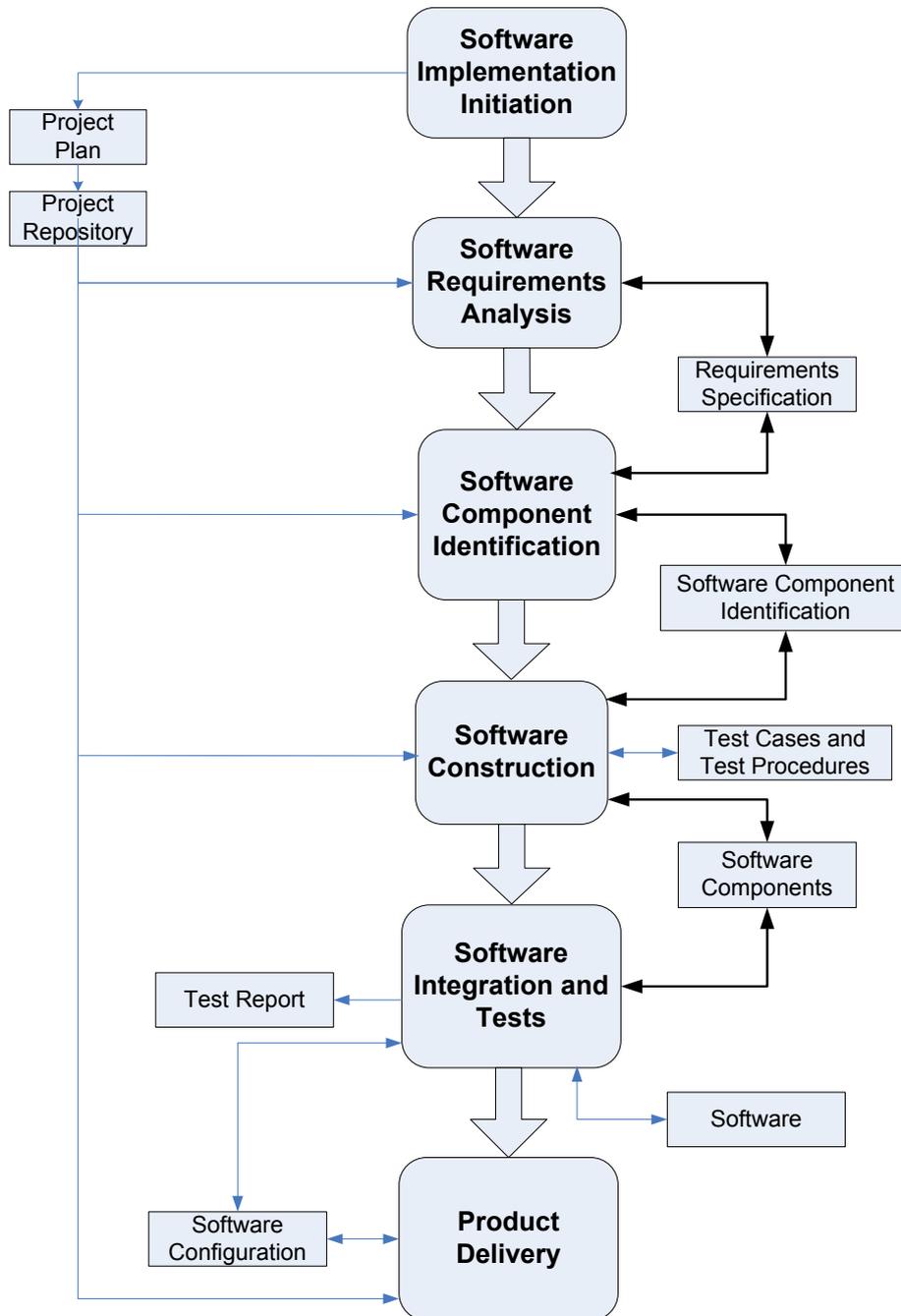
Figure 4 — Software Implementation process diagram

7.7.1 SI activities

The Software Implementation Process has the following activities:

— SI.1 Software Implementation Initiation

— SI.2 Software Requirements Analysis

— SI.3 Software Component Identification

— SI.4 Software Construction

— SI.5 Software Integration and Tests

— SI.6 Product Delivery

SI.1 Software Implementation initiation (SI.O1)

The Software Implementation Initiation activity ensures that the *Project Plan* established in Project Planning activity is committed to by the Work Team. The activity provides:

— Review of the *Project Plan* by the Work Team to determine task assignment.

— An implementation environment established.

Table 15 — SI.1 task list

Role	Task List	Input Products	Output Products
PM WT	SI.1.1 Review the current Project Plan with the Work Team members in order to achieve a common understanding and get their engagement with the project.	*Project Plan*	*Project Plan [reviewed]*
WT	SI.1.2 Set or update the implementation environment.	*Project Plan [reviewed]*	

SI.2 Software requirements analysis (SI.O2, SI.O6, SI.O7)

The Software Requirements Analysis activity analyzes the agreed customer requirements and establishes the validated project software requirements. The activity provides:

— Work Team review of the *Project Plan* to determine task assignment.

— Elicitation, analysis and specification of customer's requirements.

— Agreement on the customer requirements.

— Verification and validation of requirements.

Table 16 — SI.2 task list

Role	Task List	Input Products	Output Products
PM WT	SI.2.1 Assign tasks to the Work Team members in accordance with their role, based on the current *Project Plan.*	*Project Plan [reviewed]*	
WT CUS	SI.2.2 Document or update the *Requirements Specification.* Identify and consult information sources (customer, users, previous systems, documents, etc.) in order to get new requirements. Gather and analyze the identified requirements to determinate the scope and feasibility.	*Project Plan* - *Product Description*	*Requirements Specification [verified]*

Role	Task List	Input Products	Output Products
	Verify the correctness and testability of the *Requirements Specification* and its consistency with the *Product Description*. Generate or update the *Requirements Specification.*		
CUS	SI.2.3 Validate and obtain approval of the *Requirements Specification* Validate that *Requirements Specification* satisfies needs and agreed upon expectations, including the user interface usability.	*Requirements Specification [verified]*	*Requirements Specification [validated]*

SI.3 Software component identification (SI.O3, SI.O6, SI.O7)

The Software Component Identification activity transforms the software requirements to the architecture of system software components. The activity provides:

— Work Team review of the *Project Plan* to determine task assignment.

— Identify software components and associated interfaces.

Table 17 — SI.3 task list

Role	Task List	Input Products	Output Products
PM WT	SI.3.1 Assign tasks to the Work Team members related to their role according to the current *Project Plan.*	*Project Plan*	
WT	SI.3.2 Understand Requirements Specification.	*Requirements Specification*	
WT	SI.3.3 Document or update the *Software Component Identification.* Analyze the *Requirements Specification* to generate the components, its arrangement in subsystems and software components defining the internal and external interfaces. Provide the detail of software components and their interfaces to allow the construction in an evident way.	*Requirements Specification*	*Software Component Identification*

SI.4 Software construction (SI.O4, SI.O6, SI.O7)

The Software Construction activity develops the software code and data from the *Software Component Identification in the SI.3*. The activity provides:

— Work Team review of the *Project Plan* to determine task assignment.

— Understand the identified *Software Components*

— *Test Cases and Test Procedures* for unit and integration testing.

— Coded *Software Components* and applied unit tests.

Table 18 — SI.4 task list

Role	Task List	Input Products	Output Products
PM WT	SI.4.1 Assign tasks to the Work Team members related to their role, according to the current *Project Plan.*	*Project Plan*	
WT	SI.4.2 Understand the *Software Component Identification*	*Software Component Identification*	
WT	SI.4.3 Construct or update *Software Components.*	*Software Component Identification*	*Software Components*
WT	SI.4.4 Establish or update *Test Cases and Test Procedures* for unit and integration testing based on *Requirements Specification* and *Software Component Identification.* Customer provides testing data, if needed.	*Requirements Specification [validated]* *Software Component Identification*	*Test Cases and Test Procedures*
WT	SI.4.5 Test the Software Components. Correct the defects found until successful unit test is achieved.	*Test Cases and Test Procedures* *Software Components*	*Software Components* *[unit tested]*

SI.5 Software integration and tests (SI.O5, SI.O6, SI.O7)

The Software Integration and Test activity ensures that the integrated software components satisfy the software requirements. The activity provides:

— Work Team review of the *Project Plan* to determine task assignment.

— Understanding of *Test Cases and Procedures* and the integration environment.

— Integrated *Software Components*, corrected defects and documented results.

Table 19 — SI.5 task list

Role	Task List	Input Products	Output Products
PM WT	SI.5.1 Assign tasks to the work team members related to their role, according to the current *Project Plan.*	*Project Plan*	

Role	Task List	Input Products	Output Products
WT	SI.5.2 Understand *Test Cases and Test Procedures.* Set or update the testing environment.	*Test Cases and Test Procedures*	
WT	SI.5.3 Integrates the *Software* using *Software Components* and updates Test Cases and Test Procedures for integration testing, as needed.	*Software Components* *Test Cases and Test Procedures*	*Software* *Test Cases and Test Procedures [updated]*
WT	SI.5.4 Perform *Software* tests using *Test Cases and Test Procedures* for integration and document results in *Test Report*.	*Software* *Test Cases and Test Procedures*	*Software [tested]* *Test Report*
WT	SI.5.5 Correct the defects found until successful test is achieved.	*Software,* *Test Report.* *Test Cases and Test Procedures*	*Software [corrected]* *Test Report [defects eliminated]*
WT	SI.5.6 Incorporate the *Requirements Specification* and *Software—to the Software Configuration.*	*Requirements Specification* *Software*	*Software Configuration* *–Requirements Specification* *- Software*

SI.6 Product delivery (SI.O6, SI.O7)

The Product Delivery activity provides the integrated software product to the Project Manager and support for delivery. The activity provides:

⎯ Work Team review of the Project Plan to determine task assignment.

⎯ Delivery of the software product and applicable documentation in accordance with the Project Plan.

Table 20 — SI.6 task list

Role	Task List	Input Products	Output Products
PM WT	SI.6.1 Assign tasks to the work team members related to their role, according to the current *Project Plan.*	*Project Plan*	
WT	SI.6.2 Review for understandability the *Software Configuration*.	*Software Configuration*	
PM WT	SI.6.3 Perform delivery to Project Manager and support delivery according to the *Project Plan*.	*Project Plan,* *Software Configuration*	*Software Configuration [ready to be delivered]*

7.7.2 SI incorporation to the *Project Repository*

The list of products to be saved in *Project Repository*;

Table 21 — SI repository products

Product
Requirements Specification
Software Component Identification
Test Cases and Test Procedures
Software Components
Software
Test Report

8 Roles

This is an alphabetical list of the roles, its abbreviations and suggested competencies description. This list is showed as a four-column table for presentation purpose only.

Table 22 — Roles

	Role	Abbreviation	Competency
1.	Customer	CUS	Knowledge of the Customer processes and ability to explain the Customer requirements. The Customer (representative) must have the authority to approve the requirements and their changes. The Customer includes user representatives in order to ensure that the operational environment is addressed. Knowledge and experience in the application domain.
2.	Project Manager	PM	Leadership capability with experience making decisions, planning, personnel management, delegation and supervision, finances and software development.
3.	Work Team	WT	Knowledge and experience according to their roles on the project.

9 Product description

This is an alphabetical list of the input, output and internal process products, its descriptions, possible status and the source of the product. The source can be another process or an external entity to the project, such as the Customer. This list is showed as a four-column table for presentation purpose only. Product items in the following tables are based on ISO/IEC 15289 Information Items with some exceptions.

The product status gives the information to the project team about the type of work (tasks) already done on the product (for example: evaluated, verified, tested, baselined). This information can be used to start next tasks which can use the product as an input. Some products have no status assigned because they are only informative and they do not change the content (for example: Acceptance Record, Correction Register, Project Repository Backup, Verification/Validation Results).

Table 23 — Product Descriptions

	Name	Description	Source
1.	*Acceptance Record*	Documents the customer acceptance of the deliverables of the project. It may have the following characteristics: - Record of the receipt of the delivery - Identifies the date received - Identifies the delivered elements - Record of customer verification of the deliverables according to the *Customer agreement*. - Identifies any open issues (if applicable) - Signed by receiving Customer	Project Management
2.	*Change Request*	Identifies a software, or documentation problem or desired improvement, and requests modifications. It may have the following characteristics: - Identifies purpose of change - Identifies request status (new, agreed, rejected) - Identifies requester contact information - Impacted system(s) - Impact to operations of existing system(s) defined - Impact to associated documentation defined - Criticality of the request, date needed by The applicable statuses are: accepted and tracked.	Software Implementation Customer Project Management
3.	*Meeting Record*	Records the agreements established with Customer and/or Work Team. It may have the following characteristics: - Purpose of meeting - Attendees - Date, place held - Reference to previous minutes - What was accomplished - Identifies issues raised - Any open issues - Agreements - Next meeting, if any. The applicable status is: updated.	Project Management
4.	Progress Status Record	Records the status of the project against the Project Plan. It may have the following characteristics: - Status of actual tasks against planned tasks - Status of actual results against established objectives / goals - Status of actual resource allocation against planned resources	Project Management

	Name	Description	Source
		- Status of actual cost against budget estimates - Status of actual time against planned schedule - Status of actual risk against previously identified - Record of any deviations from planned tasks and reason why. The applicable status is: evaluated.	

5.	*Project Plan*	Presents how the project processes and activities will be executed to assure the project's successful completion, and the quality of the deliverable products. It Includes the following elements which may have the characteristics as follows:
		- *Product Description*
		    o Purpose
		    o General Customer requirements
		- *Scope* description of what is included and what is not
		- *Deliverables* - list of products to be delivered to Customer
		- *Tasks, including* verification, validation and reviews with Customer and Work Team, to assure the quality of work products. Tasks may be represented as a Work Breakdown Structure (WBS).
		- *Relationship and Dependence of the Tasks*
		- *Estimated Duration* of tasks
		- *Resources* (humans, materials, standards, equipment and tools) and the schedule when the resources are needed.
		- Composition of *Work Team*
		- *Schedule of the Project Tasks,* the expected start and completion date, for each task.
		- *Estimated Effort and Cost*
		- Identification of Project Risks
		
		NOTE: may also include these additional characteristics
		- *Resources* may include the required training.
		- *Schedule of the Project Tasks* may include the relationship and dependencies of the *Tasks*.
		- *Version Control Strategy*
		    - Product repository tools or mechanism identified
		    - Location and access mechanisms for the repository specified
		    - Version identification and control defined
		    - Backup and recovery mechanisms defined
		    - Storage, handling and delivery (including archival and retrieval) mechanisms specified
		- *Delivery Instructions*
		    - Elements required for product release identified (i.e., hardware, software, documentation etc.)
		    - Delivery requirements
		    - Sequential ordering of Tasks to be performed
		    - Applicable releases identified
		    - Identification of all delivered Software Components with version information
		    - Identification of any necessary backup and recovery procedures
		
		The applicable status is: accepted
			Project Management

	Name	Description	Source
6.	*Project Repository*	Electronic container to store project work products and deliveries. It may have the following characteristics: - Stores project work products - Stores released deliverables products - Storage and retrieval capabilities - Ability to browse content - Listing of contents with description of attributes - Sharing and transfer of work products between work team. - Effective controls over access - Maintain work products descriptions - Recovery of archive versions of work products - Ability to report work products status - Changes to work products are tracked to *Change Requests* The applicable status is: updated.	Project Management

7.	*Requirements Specification*	Identifies the software requirements.
		It may have the following characteristics:
		- Introduction –general description of software and its use within the scope of the customer business;
		
		- Requirements description:
		  - Functionality – established needs to be satisfied by the software when it is used in specific conditions. Functionality must be adequate, accurate and safe.
		
		  - User interface – definition of those user interface characteristics that allow to understand and learn the software easily so the user be able to perform his/her tasks efficiently including the interface exemplar description;
		
		  - External interfaces – definition of interfaces with other software or hardware;
		
		NOTE: may also include these additional characteristics
		
		  - Reliability – specification of the software execution level concerning the maturity, fault tolerance and recovery
		
		  - Efficiency – specification of the software execution level concerning the time and use of the Resources
		
		  - Maintenance – description of the elements facilitating the understanding and execution of the future Software modifications
		
		  - Portability – description of the Software characteristics that allow its transfer from one place to other
		
		  - Design and construction limitations/constraints – needs imposed by the Customer
			Software Implementation

	Name	Description	Source
		- Interoperability – capability for two or more systems or Software Components be able to change information each other and use it - Reusability – feature of any product/sub-product, or a part of it, so that it can be used by several users as an end product, in their own software development, or in the execution of other software products - Legal and regulative – needs imposed by laws, regulations, etc. Each requirement is identified, unique and it is verifiable or can be assessed. The applicable statuses are: verified, and validated.	
8.	*Software*	Software item (software source and executable code) for a Customer, constituted by a collection of integrated *Software Components*. The applicable status is: tested	Software Implementation
9.	*Software Component*	A set of related code units. The applicable statuses are: unit tested	Software Implementation
10.	*Software Configuration*	A uniquely identified and consistent set of software products including: - *Requirements Specification* - *Software* NOTE: may also include these additional characteristics - *Software Design* - *Traceability Record* - *Software Components* - *Test Cases and Test Procedures* - *Test Report* - *Product Operation Guide* - *Software User Documentation* - *Maintenance Documentation* The applicable statuses are: ready to be delivered and accepted.	Software Implementation
11.	*Software Component Identification*	Textual and graphical information on the software structure. This structure may include the following parts: Describes the overall *Software* structure: - Identifies the required *Software Components* - Identifies the relationship between *Software Components*	Software Implementation

12.	*Statement of Work*	Description of work to be done related to software development. It may Include:
		- Product Description
		   o Purpose
		   o General Customer requirements
		- Scope description of what is included and what is not
		- Deliverables list of products to be delivered to Customer
			Customer

	Name	Description	Source
		NOTE: may also include this additional characteristic - Objectives of the project The applicable status is: reviewed.	

13.	*Test Cases and Test Procedures*	Elements needed to test code. Test Case may include:
		  - Identifies the test case
		  - Test items
		  - Input specifications
		  - Output specifications
		  - Environmental needs
		  - Special procedural requirements
		  - Interface dependencies
		
		Test Procedures may include:
		  - Identifies: test name, test description and test completion date
		  - Identifies potential implementation issues
		  - Identifies the person who completed the test procedure
		  - Identifies prerequisites
		  - Identifies procedure steps including the step number, the required action by the tester and the expected results
			Software Implementation

14.	*Test Report*	Documents the tests execution. It may include:
		
		  - A summary of each defect
		  - Identifies the related test case
		  - Identifies the tester who found each defect
		  - Identifies the affected function(s) for each defect
		  - Identifies the date when each defect originated
		  - Identifies the date when each defect was resolved
		  - Identifies the person who resolved each defect
		
		NOTE: may also include this additional characteristic
		  - Identifies the severity for each defect
			Software Implementation

10 Software tools

10.1 Project Management process

Software tools that could be used to perform project management process activities.

Table 24 — Project management tools

Activity	Resource List
Project Planning	Project planning tool(s)
Project Plan Execution	Project scheduling tool(s)
Project Assessment and Control	Resource planning tool(s)
	Project estimation tool(s)

Project Closure	Quality planning and control tool(s)
	Process repository management tool(s)
	Configuration management tool(s)
	Risk analysis and management tool(s)

10.2 Software Implementation process

Software tools that could be used to perform software implementation process activities.

Table 25 — Software Implementation tools

Activity	Resource List
Software Implementation Initiation Software Requirements Analysis Software Component Identification Software Construction Software Integration and Tests Product Delivery	Configuration management tool(s) Change management tool(s) Traceability management tool(s) Software Design tool(s) Integrated development environment tool(s) General documentation tool(s) General communication and retrieval tool(s) General security management tool(s)
Software Requirements Analysis	Requirement analysis and specification support tool(s) Quality function deployment tool(s)
Software Construction	Source code configuration management tool(s) Compiler(s) Static/dynamic analyzer(s) Debugger tool(s) Software repository management tool(s) Construction resource database/server(s)
Software Integration and Tests	Test support tool(s) Automated test tool(s) Anomaly tracking tool(s) Error/Problem tracking tool(s) Verification and validation support tool(s) Security inspection tool(s)

Annex A
(informative)

Deployment Packages

In order to facilitate the implementation, by VSEs, of a Profile, a set of Deployment Packages are available. A deployment package is a set of artefacts developed to facilitate the implementation of a set of practices, of the selected framework, in a VSE. But, a deployment package is not a complete process reference model. Deployment packages are not intended to preclude or discourage the use of additional guidelines that VSEs find useful.

The elements of a typical deployment package are: technical description, relationships with ISO/IEC 29110, key definitions, detailed description of processes, activities, tasks, steps, roles and products, template, checklist, example, references and mapping to standards and models, and a list of tools. The mapping is only given as information to show that a Deployment Package has explicit links to Part 5, ISO standards, such as ISO/IEC 12207, or models such as the CMMI-DEV developed by the Software Engineering Institute. Hence by deploying and implementing a package, a VSE can see its concrete step to achieve or demonstrate coverage to Part 5. Deployment Packages are designed such that a VSE can implement its content, without having to implement the complete framework at the same time. The table of content of a deployment package is illustrated in table A-1.

Table A.1 — Table of Content of a Deployment Package

1. Technical Description
   Purpose of this document
   Why this Topic is important?
2. Relationships with ISO/IEC 29110
3. Key Definitions
4. Overview of Processes, Activities, Tasks, Roles and Products
5. Description of Processes, Activities, Tasks, Steps, Roles and Products
   Role Description
   Product Description
   Artefact Description
6. Template(s)
7. Example(s)
8. Checklist(s)
9. Tools
10. References to Other Standards and Models (e.g. ISO 9001, ISO/IEC 12207, CMMI-DEV)
11. References
12. Evaluation Form

For the Entry VSE Profile, a set of Deployment Packages are available, at no cost, on the Internet:

a) Project Management

b) Software Implementation

c) Self-Diagnostic, a method that VSEs can use by themselves to analyze their software processes without knowing the process terminology (i.e., no need to know exactly the meaning of process, procedure, activity, task, practice). The questions of the self-diagnostic method relate to problems that VSEs may encounter in their day-to-day operation.

Bibliography

[1]   ISO/IEC 12207:2008, *Systems and Software Engineering — Software Life Cycle Processes*

[2]   ISO/IEC 15289:2011, *Systems and software engineering — Content of life-cycle information products (documentation)*

[3]   ISO/IEC TR 29110-1, *Software engineering — Lifecycle Profiles for Very Small Entities (VSEs) — Part 1: Overview*

[4]   ISO/IEC 29110-4-1, *Software engineering — Lifecycle Profiles for Very Small Entities (VSEs) — Part 4-1: Profile specifications — Generic profile group*

[5]   OECD. *SME and Entrepreneurship Outlook, 2005 Edition*. Organization for Economic Co-Operation and Development, Paris, 2005

ICS 35.080

Price based on 32 pages