**Deployment Package**

**Self-Assessment**

**Systems Engineering Basic Profile**

**Notes:**

This document is the intellectual propriety of its author’s organization. However, information contained in this document is free of use. The distribution of all or parts of this document is authorized for non-commercial use as long as the following legal notice is mentioned:

© International Council on Systems Engineering (INCOSE)

Commercial use of this document is strictly forbidden. This document is distributed in order to enhance exchange of technical and scientific information.

This material is furnished on an “as-is” basis. The author(s) make(s) no warranties of any kind, either expressed or implied, as to any matter including, but not limited to, warranty of fitness for purpose or merchantability, exclusivity, or results obtained from use of the material.

The process described in this Deployment Package is not intended to preclude or discourage the use of additional processes that Very Small Entities may find useful.

|  |  |
| --- | --- |
| **Author(s)** | *Name, - INCOSE, name of country* |
| **Editors** | Claude Y. LAPORTE – École de technologie supérieure (ÉTS), (Canada)  Martin GEISREITER – INCOSE, Germany  *To be completed* |
| **Creation date** | *Day-Month-year (e.g. 12 June 2012)* |
| **Last update** | *Day-Month-year* |
| **Version** | *X.X* |

Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date**  (dd/mm/yyyy) | **Version** | **Description** | **Author** |
| *<dd/mmm/yy>* | *<x.x>* | Initial Release | *<name>* |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Abbreviations/Acronyms

|  |  |
| --- | --- |
| **Abre./Acro.** | **Definitions** |
| DP | Deployment Package - a set of artefacts developed to facilitate the implementation of a set of practices, of the selected framework, in a Very Small Entity. |
| INCOSE | International Council on Systems Engineering (<http://www.incose.org>) |
| ISO | International Organization for Standardization (<http://www.iso.org>) |
| SW | Software |
| SY | System |
| TR | Technical Report |
| VSE | Very Small Entity – enterprise, organization, department or project having up to 25 people. |
| VSEs | Very Small Entities |
| V&V | Verification and Validation |
| PMBOK | Project Management Body of Knowledge, <http://www.pmi.org/> |
| *<details>* | *<details>* |
|  |  |

Table of Contents

[1. Technical Description 4](#_Toc355185847)

[Purpose of this document 4](#_Toc355185848)

[Why is Systems Engineering important? 4](#_Toc355185849)

[Why is cooperation between Systems Engineering and Project Management important? 4](#_Toc355185850)

[Why this topic is Important? 6](#_Toc355185851)

[2. Definitions 7](#_Toc355185852)

[Generic Terms 7](#_Toc355185853)

[Specific Terms 7](#_Toc355185854)

[3. Relationships with ISO/IEC 29110 8](#_Toc355185855)

[4. Description of Activities, Tasks, Steps, Roles and Products 10](#_Toc355185856)

[<Task Title> 10](#_Toc355185857)

[Role Description 12](#_Toc355185858)

[Product Description 12](#_Toc355185859)

[Artefact Description 13](#_Toc355185860)

[5. Template 14](#_Toc355185861)

[6. Example of Activity Lifecyle 15](#_Toc355185862)

[7. Checklist 16](#_Toc355185863)

[8. Tool 17](#_Toc355185864)

[9. Reference to other Standards and Models 19](#_Toc355185865)

[ISO 9001 Reference Matrix 19](#_Toc355185866)

[ISO/IEC 12207 Reference Matrix 19](#_Toc355185867)

[CMMI for Development Reference Matrix 19](#_Toc355185868)

[10. References 20](#_Toc355185869)

[11. Evaluation Form 21](#_Toc355185870)

# 1. Technical Description

## Purpose of this document

A Deployment Package (DP) is a set of artifacts developed to facilitate the implementation of a set of practices in a Very Small Entity (VSE). A DP is not a process reference model (i.e. it is not prescriptive). The elements of a typical DP are: roles and products, description of processes, activities, tasks, template, checklist, reference to standards, etc.

This Deployment Package (DP) supports the Basic Profile as defined in ISO/IEC TR 29110-5-6-2, the Management and engineering guide [ISO/IEC 29110]. The Basic Profile is one profile of the Generic profile group. The Generic profile group is applicable to VSEs that do not develop critical systems. The Generic profile group is composed of 4 profiles: Entry, Basic, Intermediate and Advanced. The Generic profile group does not imply any specific application domain. The Basic profile is targeted to VSEs working on one project at a time.

The Basic profile is composed of two processes: the Project Management Process and the System Definition and Realization Process.

The processes, activities and tasks described in this DP are consistent with those listed in ISO/IEC TR 29110 5-6-2 Systems Engineering — Lifecycle Profiles for Very Small Entities (VSEs) — Part 5-6-2: Management and engineering guide – Generic profile group: Basic profile.

The INCOSE Systems Engineering Handbook [INCOSE] has been used to develop this DP. The INCOSE Handbook is consistent with ISO/IEC 15288:2008 – *Systems and software engineering – System life cycle processes* [ISO 15288].

Information contained in this DP is applicable to VSEs that do not develop critical products that require intense verification and validation (V&V) activities. Those projects could use of the appropriate standards and guides (e.g. ANSI/GEIA EIA-632, MIL-STD-499, etc.)

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

The content of this document is entirely *informative*.

Once published by ISO, ISO/IEC TR 29110-5-6-2 will be available at no cost on the following ISO site: <http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html>

## Why is Systems Engineering important?

The way to effective Systems Engineering (SE) is not “in the direction of formal, formidable, massive documentation” [Chase]. Systems Engineering is a perspective, a process, and a profession [INCOSE]. SE has an iterative nature that supports learning and continuous improvement. SE has a horizontal orientation which means SE is a mechanism to establish agreements for the creation of products and services in a web of contractors and subcontractors. Therefore SE is the link between contractors, and PM, and the organizational parts of enterprises and single technical disciplines (e.g. Software, mechanics, HMI, EMC, etc.).

## Why is cooperation between Systems Engineering and Project Management important?

Systems engineers and program managers bring unique skills and experiences to the programs on which they work. There is also a “shared space” (PM/SE) where program managers and systems engineers collaborate to drive the program team’s performance and success. Therefore they have to collaborate.

Figure 1 shows a concept how systems engineering (SE) and project management (PM) might relate to each other. The basis for this concept is the project lifecycle as proposed by *ISO 21500 – Guidance on project management*. But it’s too simple just to consider the pure project time span for a product development. SE has to consider the whole life cycle of a product in the product concepts until product disposal. Therefore SE has to contribute in all project control activities and provide relevant inputs.



Figure 1 Overview of a concept for SE – PM cooperation

Because a deployment package is not a complete process reference model a VSE might need guidance about how they might perform a project.

To consider the idea for the ISO/IEC TR 29110 simplified technical processes have been defined (see the 9 coloured blocks in Figure 2). Each of these blocks consists of business aspects and technical aspects. Just the degree of involvement for PM and SE changes. Interface management or requirements engineering are commonly understand as SE activities. But they are also influenced by business aspects, enterprise interests or simply by available resources which are more in the PM domain. Therefore the addressed technical processes in Figure 2 might be understood as common (PM&SE) activities.

Configuration management (CM) might be understood as an enterprise oriented task and used in every project. The activities of CM should start with the earliest project activities (the first idea for a project) and will not end with a project. The stored information must be available after a project is finished for several purposes (e.g. following project, legal issues, etc.).

Each of the technical process blocks includes activities which might be performed in different project phases. Figure 2 shows an example to map project process steps to single technical processes. The details for the technical processes are described in different DP packages.

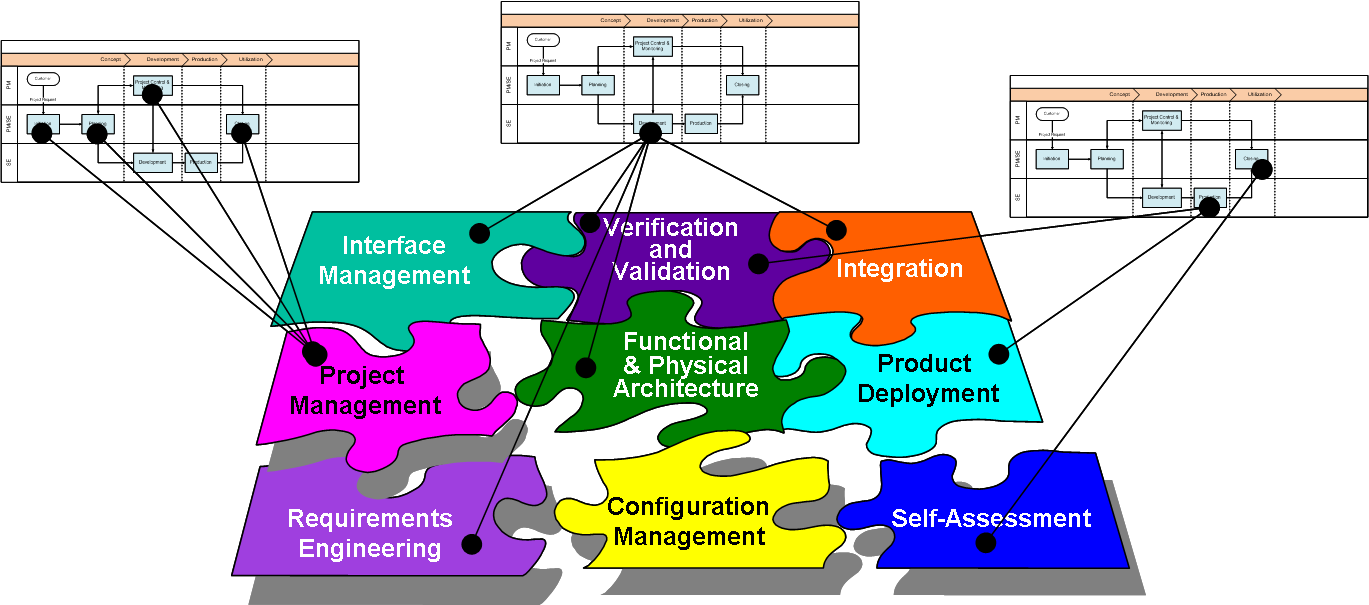


Figure 2 DP structure and linkage to project steps

## Why this topic is Important?

*<Add text here about this DP>*

# 2. Definitions

In this section, the reader will find two sets of definitions. The first set defines the terms used in all Deployment Packages, i.e. generic terms. The second set of terms used in this Deployment package, i.e. specific terms.

## Generic Terms

***Process:*** set of interrelated or interacting activities which transform inputs into outputs [ISO/IEC 12207].

***Activity:*** a set of cohesive tasks of a process [ISO/IEC 12207].

***Task:*** required, recommended, or permissible action, intended to contribute to the achievement of one or more outcomes of a process[ISO/IEC 12207].

***Sub-Task:*** When a task is complex, it is divided into sub-tasks.

***Step:*** one element (numbered list item) in a procedure that tells a user to perform an action (or actions) [ISO/IEC 26514]. In a deployment package, a taskis decomposed in a sequence of steps.

***Role***: a defined function to be performed by a project team member, such as testing, filing, inspecting, coding. [ISO/IEC 24765]

***Product:*** piece of information or deliverable that can be produced (not mandatory) by one or several tasks. *(e. g. design document, source code)*.

***Artefact:*** information, which is not listed in ISO/IEC 29110 Part 5, but can help a VSE during the execution of a project.

***System:*** combination of interacting elements organized to achieve one or more stated purposes. [ISO/IEC 15288:2008]

***Software:*** all or part of the programs, procedures, rules, and associated documentation of an information processing system. [ISO/IEC 2382-1]

## Specific Terms

*<details>*

# 3. Relationships with ISO/IEC 29110

This deployment package covers the activities related to <topic> of the ISO Technical Report ISO/IEC 29110 Part 5-1-2 for Very Small Entities (VSEs) – Basic VSE Profile [ISO/IEC 29110].

In this section, the reader will find a list of Project Management (PM) and <System or Software> Implementation process, activities, tasks and roles from Part 5 that are directly related to this topic. This topic is described in details in the next section.

* **Process:** <Number and Title>
* **Activity:** <Number and Title>
* **Tasks and Roles:**

|  |  |
| --- | --- |
| **Tasks** | **Roles[[1]](#footnote-1)** |
| * <Number and Title> | Abbreviation |

* **Process:** <Number and Title>
* **Activity:** <Number and Title>
* **Tasks and Roles:**

|  |  |
| --- | --- |
| **Task** | **Roles** |
| * <Number and Title> | Abbreviation |

**Example:**

* ***Process:*** *4.3[[2]](#footnote-2) System/Software Implementation*
* ***Activity:*** *4.3.8.2 System/Software Requirements Analysis*
* ***Tasks and Roles:***

|  |  |
| --- | --- |
| ***Tasks*** | ***Roles*** |
| *SI.2.2 Document or update the Requirements Specification.* | *AN, CUS* |

* ***Process:*** *4.2 Project Management Process (PM)*
* ***Activity:*** *4.2.8.3 Project Assessment and Control*
* ***Tasks and Roles:***

|  |  |
| --- | --- |
| ***Task*** | ***Roles*** |
| *PM.3.2 Establish actions to correct deviations or problems and identified risks concerning the accomplishment of the plan, as needed, document them in Correction Register and track them to closure.* | *PM, TL, WT* |

# 4. Description of Activities, Tasks, Steps, Roles and Products

## <Task Title>

|  |  |
| --- | --- |
|  | |
| ***Objectives:*** | *<details>* |
| ***Rationale:*** | *<details>* |
| ***Roles:*** | *<Abbreviation>* |
| *<Abbreviation>* |
| ***Products:*** | *<Title of Product>* |
| *<details>* |
| ***Artifacts:*** | *<details>* |
| *<details>* |
| ***Steps:*** | 1. *<details>* |
| 2. *<details>* |
| ***Step Description:*** | ***Step 1. <****Active verb + Noun>*  Description of the step, input/output, form used, etc.  ***Step 2. <****Active verb + Noun>* |

**Example:**

***Process: 4.3 System/Software Implementation***

*The purpose of the System/Software Implementation process is the systematic performance of the analysis, design, construction, integration and tests activities for new or modified System/software products according to the specified requirements.*

***Activity: 4.3.8.2 System/Software Requirements Analysis***

*The System/Software Requirements Analysis activity analyzes the agreed customer’s requirements and establishes the validated project requirements. The activity provides:*

|  |  |
| --- | --- |
| ***Tasks*** | ***Roles[[3]](#footnote-3)*** |
| *SI.2.2 Create or update the Requirements Specification.* | *TL, WT* |

**Example:**

***Requirements identification***

|  |  |
| --- | --- |
|  | |
| ***Objectives:*** | *To clearly define the scope of the project and identify the key requirements of the system.* |
| ***Rationale:*** | *It is important to clearly define the project scope (boundaries) and to identify key functionalities of the future system with the customer to avoid problems like forgotten key functionalities or requirements creep.* |
| ***Roles:*** | *AN - Analyst* |
| *CUS- Customer* |
| *PM- Project Manager* |
| *WT Work Team* |
| ***Products:*** | *Requirements Specification* |
| ***Artifacts:*** | *Use Cases – scenarios (depending on the implementation)* |
| ***Steps:*** | *Step 1. Collect information about the application domain (e.g. finance, medical)* |
| *Step 2. Identify project’s scope* |
| ***Step Description:*** | ***Step 1. Collect information about the domain****:*  *During this Step, analyst captures the key concepts of the business domain of the customer. The customer assists the analyst by giving him all the information (existing documentation or explanation) that will facilitate this understanding.*  *Key concepts are listed in a glossary section in the System/Software Requirements Specification Document outline document.*  ***Step 2. Identify project’s scope***  *System/Software analyst, helped by* *the person in charge of the contractual aspects of the project (sales manager) clearly identifies main functionalities that are included in the project scope.*  *Tips: Identifying functionalities that are OUT of scope is also very valuable to clarify differences of understanding with your customers.* |

## Role Description

This is an alphabetical list of the roles, abbreviations and list of competencies as defined in the engineering and management guide of ISO/IEC 29110..

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Role*** | ***Abbreviation*** | ***Competency*** |
| 1. |  |  |  |
| 2. |  |  |  |
| 3. |  |  |  |

**Example:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Role*** | ***Abbreviation*** | ***Competency*** |
| *1.* | *Analyst* | *AN* | *Knowledge and experience eliciting, specifying and analyzing the requirements.*  *Knowledge in designing user interfaces and ergonomic criteria.*  *Knowledge of the revision techniques and experience on the system/software development and maintenance.*  *Knowledge of the editing techniques and experience on the system/software development and maintenance.* |

## Product Description

This is an alphabetical list of the input, output and internal process products, its descriptions, possible states and the source of the product.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Name** | **Description** | **Source** |
| 1. |  |  |  |
| 2. |  |  |  |
| 3. |  |  |  |

**Example:**

|  | **Name** | **Description** | **Source** |
| --- | --- | --- | --- |
| *1.* | *Change Request* | *It may has the following characteristics:*  *Identifies purpose of change*  *Identifies request status (new, accepted, 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: initiated, evaluated and accepted.* | *System/Software Implementation*  *Customer*  *Project Management* |

## Artefact Description

This is an alphabetical list of the artifacts that could be produced to facilitate the documentation of a project. The artifacts are not required by Part 5, they are optional.

|  | **Name** | **Description** |
| --- | --- | --- |
| 1. | *<details>* | *<details>* |

# 5. Template

*<Template(s) is/are provided to help VSE produce a better product> <details>*

**Example:**

*SRS Template Table of Content –Basic List of Requirements*

*To be used in an Excel sheet structured, for example, as:*

|  |  |  |  |
| --- | --- | --- | --- |
| *ID* | *Requirement* | *Description* | *Priority* |
|  |  |  |  |

*SRS Template Table of Content –Adapted from IEEE 830*

***1. Introduction***

*1.1 Purpose*

*1.2 Document conventions*

*1.3 Intended audience*

*1.4 Additional information*

*1.5 Contact information/SRS team members*

*1.6 References*

# 6. Example of Activity Lifecyle

*This section provides, for this topic, a graphical representation of a lifecycle. The example is provided to help the reader implement his own lifecycle fitting his IT project’s context and constraints.*

*<details>*

**Example:**

Example of Requirement Practices Lifecycle

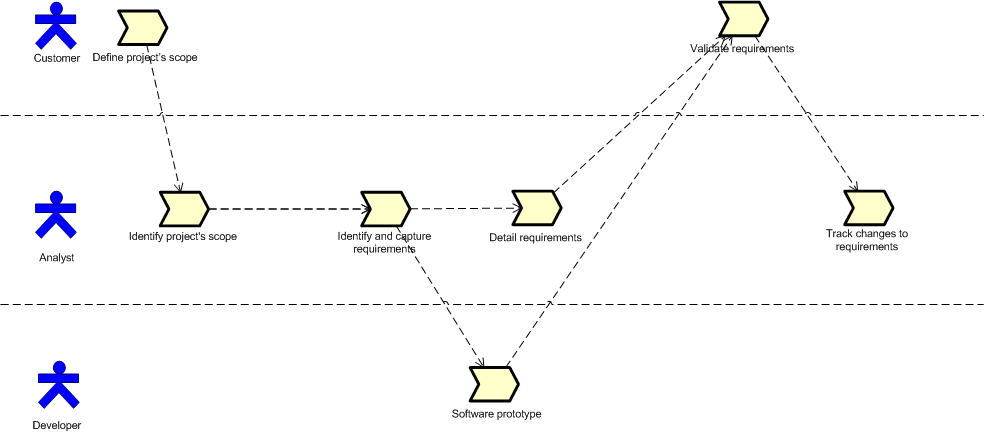


Figure X Example of Requirement Practices Lifecycle

# 7. Checklist

*<Checklist(s) is/are provided to help VSE produce a better product> <details>*

**Example**:

*Requirements Checklist*

*This requirements checklist is adapted from [Constr07]*

|  |  |
| --- | --- |
| ***RS 1 Testable*** | *Are all the requirements are verifiable (objectively)?* |
| ***RS 2 CompleteRS 2 Complete*** | *Are the requirements complete?* |
| ***RS 3 ClearUnambiguous Unambiguous RS 10 Unambiguous*** | *Are all the requirements clear to all reviewers? (e.g. tester) SRS must contain requirements statements that can be interpreted in one way only.SRS must contain requirements statements that can be interpreted in one way only* |

# 8. Tool

*<List the reference(s) (e.g. address of web site) to tool(s) that can help implement this deployment package. If a tool is simple to describe, you may provide the description to use it in this section. See the example below>*

*<details>*

**Example:**

***Traceability Tool***

*Requirements traceability should:*

|  |  |
| --- | --- |
|  | * *Ensure traceability for each level of decomposition performed on the project. In particular:*   + *Ensure that every lower level requirement can be traced to a higher level requirement or original source*   + *Ensure that every design, implementation, and test element can be traced to a requirement*   + *Ensure that every requirement is represented in design and implementation*   + *Ensure that every requirement is represented in testing/verification* * *Ensure that traceability is used in conducting impact analysis of requirements changes on project plans, activities and work products* * *Be maintained and updated as changes occur.* * *Be consulted during the preparation of Impact Analysis for every proposed change to the project* * *Be planned for, since maintaining the links/references is a labor intensive process that should be tracked/monitored and should be assigned to a project team member* * *Be maintained as an electronic document* |

**

|  |  |
| --- | --- |
| ***Instructions*** | |
| *The above table should be created in a spreadsheet or database such that it may be easily sorted by each column to achieve bi-directional traceability between columns. The unique identifiers (ID) should be assigned in a hierarchical outline form such that the lower level (i.e. more detailed) items can be traced to higher items.* | |
| *Identification Number* | *The Unique Requirement Identification (ID) where the requirement is referenced, and/or the unique identification for decomposed requirements.* |
| *Text of the need* | *The original text of the need from the customer* |
| *Text of the requirement* | *The text of the requirement* |
| *Verification Method* | *The verification method is identified (e.g. Test (T), Demonstration (D), Analysis (A), Simulation (S), Inspection (I)).* |
| *Title or ID of Use Case* | *The unique identifier of the Use Case or design component where a requirement is designed.* |
| *Title or ID of Code Module* | *The unique identifier of the system/software module where the design is realized or coded.* |
| *Verification Date* | *The date the requirement is verified (e.g. tested)* |
| *Name of person that performed the verification* | *The name of the person that performed the verification* |
| *Result of the verification* | *Result of verification (i.e. Success (S) or Failure (F))* |

# 9. Reference to other Standards and Models

This section provides references of this deployment package to selected ISO and ISO/IEC Standards and to the Capability Maturity Model IntegrationSM for Development, Version 1.3 of the Software Engineering Institute (CMMI®[[4]](#footnote-4)).

Notes:

* This section is provided for information purpose only.
* Only tasks covered by this Deployment Package are listed in each table.
* The tables use the following convention:
* Full Coverage = F
* Partial Coverage = P
* No Coverage = N

## ISO 9001 Reference Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| **Clause of ISO 9001** | **Coverage**  **F/P/N** | **Title of the Task and Step** | **Comments** |
| *<details>* | *<details>* | *<details>* |  |
|  |  |  |  |
|  |  |  |  |

## ISO/IEC 12207 Reference Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| **Clause of ISO/IEC 12207** | **Coverage**  **F/P/N** | **Title of the Task and Step** | **Comments** |
| *<details>* | *<details>* | *<details>* |  |
|  |  |  |  |
|  |  |  |  |

## CMMI for Development Reference Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| **Objective/ Practice of CMMI V1.3** | **Coverage**  **F/P/N** | **Title of the Task and Step** | **Comments** |
| *<details>* | *<details>* | *<details>* |  |
|  |  |  |  |
|  |  |  |  |

# 10. References

|  |  |
| --- | --- |
| **Key** | **Reference** |
| [ISO/IEC 12207] | ISO/IEC 12207:2008 Systems and software engineering - Software life cycle processes. |
| [ISO/IEC 15288:2008] | ISO/IEC 15288:2008 Systems and software engineering - System life cycle processes. |
| [ISO/IEC 26514] | ISO/IEC 26514, Systems and software engineering — Requirements for designers and developers of user documentation. |
| [ISO/IEC/IEEE 24765] | ISO/IEC/IEEE 24765:2010, Systems and Software Engineering - Vocabulary.  Available on line at:  <http://pascal.computer.org/sev_display/index.action> |
| [ISO/IEC 29110] | ISO/IEC 29110:2011-5-1-2 - Software Engineering — Lifecycle Profiles for Very Small Entities (VSEs) — Part 5-1-2: Management and Engineering Guide – Generic Profile Group - Basic Profile. |
| [ISO/IEC 2382-1] | ISO/IEC 2382-1:1993, Information technology — Vocabulary — Part 1: Fundamental terms. |
| *<details>* | *<details>* |

# 11. Evaluation Form

|  |
| --- |
| **Deployment Package *<Title>–* *Version <X.X>***  Your feedback will allow us to improve this deployment package, your comments and suggestions are welcomed. |
| **1. How satisfied are you with the CONTENT of this deployment package?**   *Very Satisfied*  *Satisfied*  *Neither Satisfied nor Dissatisfied*  *Dissatisfied*  *Very Dissatisfied* |
| **2. The sequence in which the topics are discussed, are logical and easy to follow?**   *Very Satisfied*  *Satisfied*  *Neither Satisfied nor Dissatisfied*  *Dissatisfied*  *Very Dissatisfied* |
| **3. How satisfied were you with the APPEARANCE/FORMAT of this deployment package?**   *Very Satisfied*  *Satisfied*  *Neither Satisfied nor Dissatisfied*  *Dissatisfied*  *Very Dissatisfied* |
| **4. Have any unnecessary topics been included? (please describe)** |
| **5. What missing topic would you like to see in this package? (please describe)**   * Proposed topic: * Rationale for new topic |
| **6. Any error in this deployment package?**   * + Please indicate:     - * Description of error :       * Location of error (section #, figure #, table #) : |
| **7. Other feedback or comments:** |
| **8. Would you recommend this Deployment package to a colleague from another VSE?**   *Definitely*  *Probably*  *Not Sure*  *Probably Not*  *Definitely Not* |

**Optional**

* Name:
* e-mail address : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Email this form to**: [claude.y.laporte@etsmtl.ca](mailto:claude.y.laporte@etsmtl.ca)

1. Roles are defined in a next section. Roles are also defined in the engineering and management guide of ISO/IEC 29110 [↑](#footnote-ref-1)
2. These numbers refer to processes, activities, tasks of the engineering and management guide of ISO/IEC 29110 [↑](#footnote-ref-2)
3. Roles are defined in a next section. Roles are also defined in the engineering and management guide of ISO/IEC 29110 [↑](#footnote-ref-3)
4. SM CMM Integration is a service mark of Carnegie Mellon University.

   ® Capability Maturity Model, CMMI are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University. [↑](#footnote-ref-4)