

InSecTT: Intelligent Secure Trustable Things



Requirements Management Guidelines

Document Type Deliverable
Document Number D1.1
Primary Author(s) Joachim Hillebrand | VIF
Document Version / Status 1.0 | Final

Distribution Level PU (public)

Project Acronym InSecTT
Project Title Intelligent Secure Trustable Things
Project Website <https://www.insectt.eu/>
Project Coordinator Michael Karner | VIF | michael.karner@v2c2.at
JU Grant Agreement Number 876038
Date of latest version of Annex I against which the assessment will be made 2020-05-20



CONTRIBUTORS

Name	Organization	Name	Organization
J. Hillebrand	VIF		

FORMAL REVIEWERS

Name	Organization	Date
Michael Jerne	NXP	2020-07-08
Lukasz Szczygielski	GUT	2020-07-24

DOCUMENT HISTORY

Revision	Date	Author / Organization	Description
0.1	2020-07-03	Joachim Hillebrand / VIF	Initial description of requirements management approach
1.0	2020-07-07	Joachim Hillebrand / VIF	Finalized document with input from Kick-off meeting and online meeting from 29/06/2020
1.1	2020-08-04	Joachim Hillebrand / VIF	Integrated input from reviews

TABLE OF CONTENTS

1	EXECUTIVE SUMMARY	4
2	OBJECTIVES	5
3	DESCRIPTION OF WORK	6
3.1	InSecTT principles as basis for requirements	6
3.2	InSecTT requirements format	8
3.3	InSecTT requirements collection	11
3.4	Requirements timeline	14
3.5	Outlook to requirements alignment	15
3.6	Outlook to progress assessment	15
4	DISSEMINATION, EXPLOITATION AND STANDARDISATION	16
5	CONCLUSIONS	17
6	REFERENCES	18
A.	ABBREVIATIONS AND DEFINITIONS	19

LIST OF FIGURES

Figure 1	Principles for mapping of building blocks to Use cases	6
Figure 2	Iterative requirements elicitation from UC and BB sides.....	7
Figure 3	Combined InSecTT requirement assigned to UCs and BBs	8
Figure 4	UC choice	9
Figure 5	BB choice.....	10
Figure 6	Implementation release choice.....	10
Figure 7	Status choice	11
Figure 8	Requirements data base on the InSecTT SharePoint.....	12
Figure 9	Pop-up menu for entering requirements	13
Figure 10	Timeline of InSecTT requirements management with 3 iterations	14

1 EXECUTIVE SUMMARY

This deliverable describes the requirements management methodology carried out in the InSecTT project and serves as a guideline for all InSecTT partners. According to the guideline, InSecTT partners are asked to provide requirements for their planned implementations and enter them into a common requirements data base on the InSecTT SharePoint platform. Requirements of application-oriented use cases and technology-oriented building blocks are handled uniformly with a common requirements format.

To support all InSecTT partners to run this process as efficient as possible, D1.1 therefore explains InSecTT principles, comprehensively describes the requirements management process as well as timelines and gives additional guidance on best practices by using the SharePoint platform.

The process described in this document is mandatory for all InSecTT partners during the execution period of the InSecTT project.

Keywords: Requirements Management, Guideline, Building Blocks, Use Cases

2 OBJECTIVES

InSecTT has set very ambitious timelines regarding the initial requirements entry and alignment process. To comply to these targets, clear and efficient guidance has to be provided to all InSecTT partners. This is the primary objective of this deliverable. Results achieved and learnings made in the predecessor projects DEWI[2] and SCOTT[3] are considered and used as a starting point in InSecTT.

This deliverable covers following objectives from WP1 described in the DoA [1]:

- Development of requirement guidelines and best practices
- Elaboration of requirements for the project
- Harmonization and tracking of requirements

As a first step, the focus is on the clear objective to quickly provide well-understandable guidance to all InSecTT partners, how to engage in the InSecTT requirements management entry process. The first and most important wave has to take place in the first three months of the project to allow a quick start of implementation activities, however, two more iterations are scheduled after one resp. two years. They are also reflected in this document.

D1.1 is based upon a guidance file in ppt-format, which has already been distributed in the beginning of July 2020 to all partners, in order to support the ambitious timing and to provide guidance even before this deliverable becomes broadly available. The content has been developed and agreed with all work package leaders and deployed to partners in two sessions in the InSecTT kick-off meeting as well as in a dedicated requirements definition meeting.

Details on the methods for alignment of requirements will be defined as part of the alignment process itself and will be reported in the subsequent deliverable D1.4 [4].

Details on the the methods for assessment of progress will be defined as part of the assessment process itself and will be reported in InSecTT deliverable D1.6 [5].

3 DESCRIPTION OF WORK

In this chapter the InSecTT principles and timelines are explained, followed by a detailed description of the requirements management process including how this is supported from a tool perspective. The description includes the requirements entry steps in detail for the first entry wave, and describes optional update cycles after year 1 and year 2.

Finally the relation to the alignment and assessment process is briefly described.

3.1 InSecTT principles as basis for requirements

The InSecTT project builds upon the following key principles:

- An agile approach with up to three implementation loops is supported
- The main requirements entry & alignment phase happens at project start (M2-4)
- Optional update cycles are in M12-14 and M24-26 (in case of iterations or major changes)
- InSecTT Use Cases rely on the availability of “fit-for-purpose” technical Building Blocks
- Use Cases (resp. related WPs) and Building Blocks (resp. related WPs) are the major source for BB-requirements
- However, requirements can come from any WP (e.g. also project level; or domain level)

The mapping principle between Use Cases and Building Blocks (not all are listed here) is shown in Figure 1. The illustrations are the same as proposed in the DoA [1] and remain unchanged considering all input at the starting phase of the project.

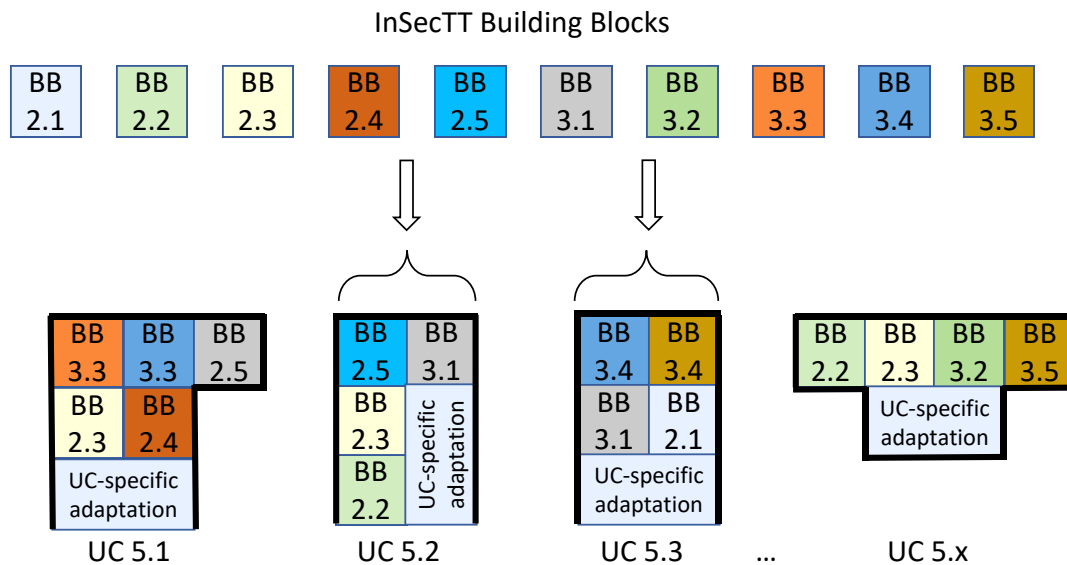


Figure 1 Principles for mapping of building blocks to Use cases

The 10 InSecTT building blocks BB2.1 to BB3.5 cover the technological developments, whereas the 16 InSecTT use cases are composed of a combination of building blocks. One use case is made up of several building blocks extended by some UC-specific adaptation content that forms the glue to the specific applications.

From a requirements point of view, both BBs and UCs cause requirements, but in slightly different nature. Note that Figure 1 only shows an illustrative example of composition and not the real structure of use cases.

Figure 2 shows one potential way how the requirements can be developed within InSecTT. Generally, the uses cases cause a need for specific features that reflects in requirements for building blocks (step 1). After the BBs are integrated (step 2), they will be integrated into the UCs (step 3), where a first iteration is completed with a demonstration (step 4). From the results, additional commercial and social requirements may be derived that flow into the next iteration loop.

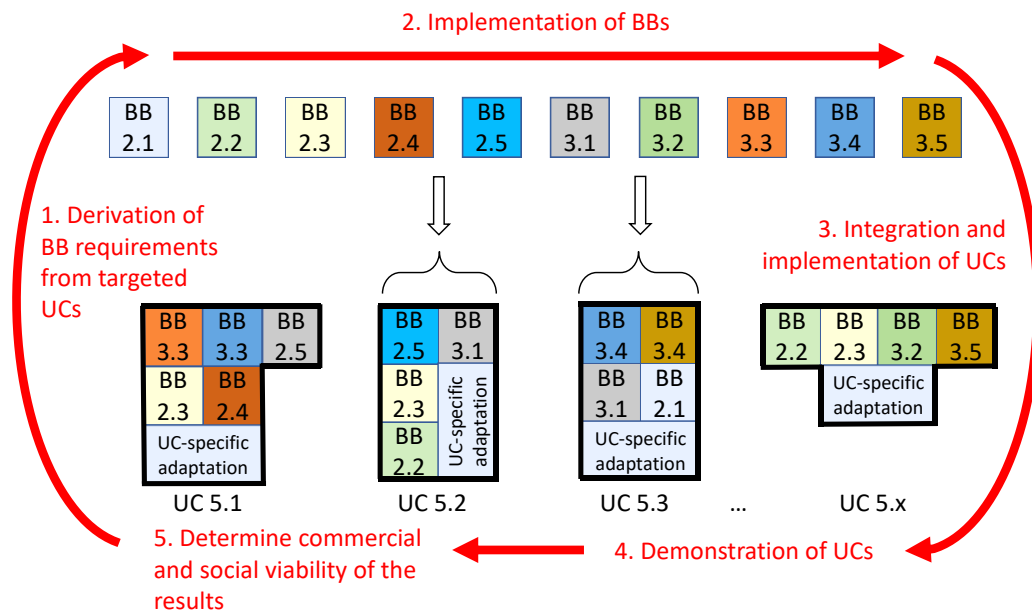


Figure 2 Iterative requirements elicitation from UC and BB sides

The two natures of a requirement can be seen with the following examples:

a.) Similar requirements from BBs and UCs

- Example for BB (technical) requirement:
 - *The AI algorithm must anonymize personal data after processing*
 - This can be mapped to T2.1 (AI on application level)
- Example for UC requirement (more concrete application):
 - *It shall not be possible to find out which person has taken the bus*
 - This can be mapped to: UC5.15 (Public Transport)

These two requirements have an overlap in their meaning, this becomes especially important, when BBs are integrated into UCs. It is recommended that requirements are formulated as UC-agnostic as possible in order to be able to map it to several UCs.

- Resulting example requirement:
 - *The AI algorithm must anonymize personal data after processing* → this formulation is already agnostic towards use cases
 - This can then be mapped to both UC5.15 (Public Transport), because it relates to people around a public bus and UC5.12 (Driver Monitoring and Distraction Detection using AI), because it also relates to drivers of cars

b.) Contradicting requirements from BBs and UCs

Example for BB (technical) requirement:

- *The AI algorithm must anonymize personal data after processing*

Example for UC requirement

- *The AI algorithm should not anonymize personal data after processing*

Potential resulting example requirement:

- *The AI algorithm should allow to set anonymization level of personal data after processing”*

It can be seen that due to the strong interconnection of BBs and UCs, a separate treatment for requirements for both of them becomes difficult. The InSecTT methodology will therefore cover combined requirements that show characteristics of both UCs and BBs.

3.2 InSecTT requirements format

Based on the principles described above, a consistent format for InSecTT requirements has been elaborated in the beginning of the project. The structure is also based on lessons learned from other European R&D projects, where similar requirements management methodologies have been deployed, like the ARTEMIS project DEWI [2] and ECSEL project SCOTT [3].

Figure 3 shows how the unified InSecTT requirements format is related to both UCs and BBs. One line denotes one requirement that is assigned to one BB on the left and one or several UCs at the top. The granularity of requirements in InSecTT targets a manageable number of requirements for the overall project. This target is set to about 500 – 600 requirements, also considering the experiences gained from the previous research projects. This is only a rough orientation as seen from the beginning of the project, the exact number might differ depending on the real requirement characteristics.

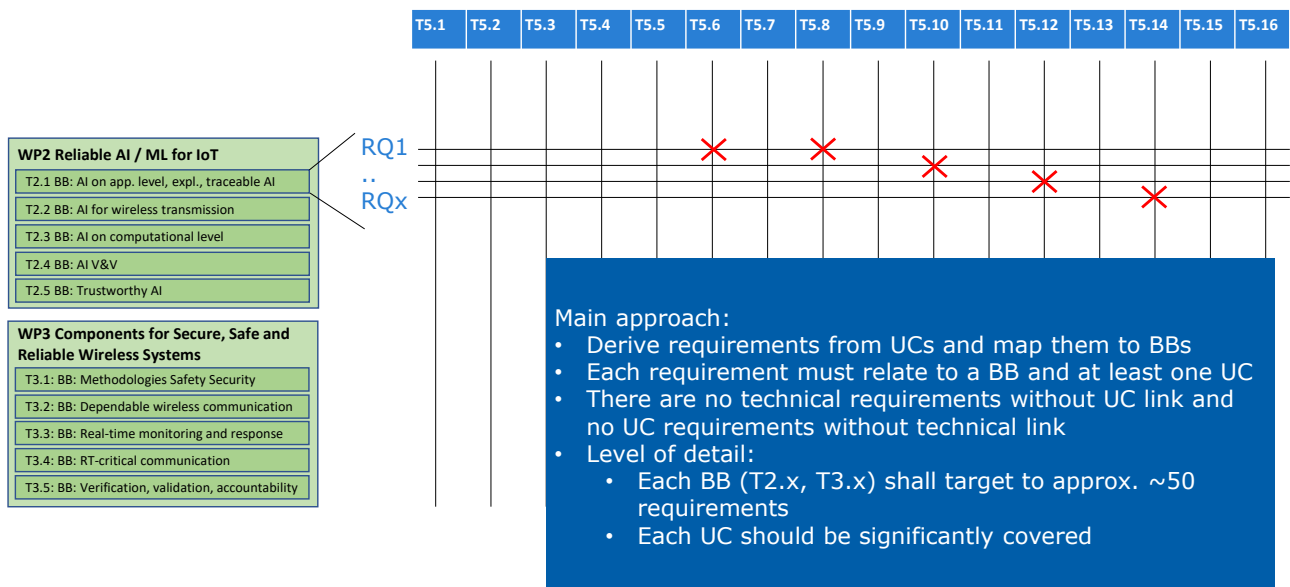
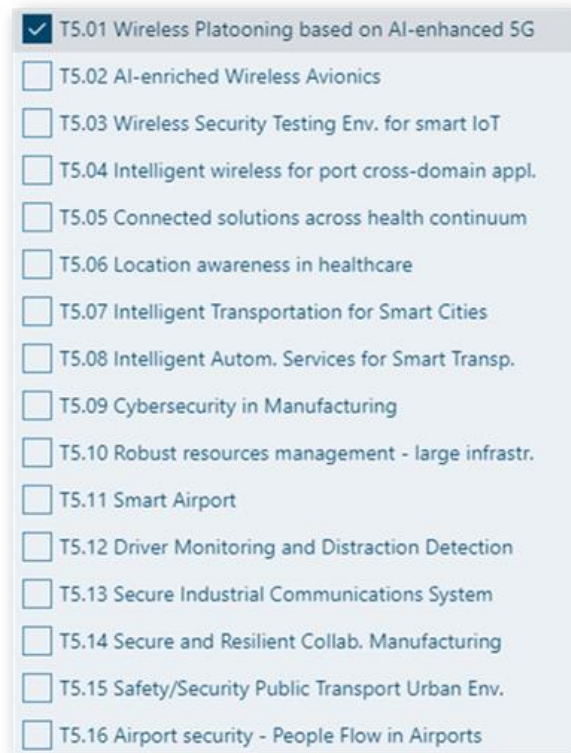


Figure 3 Combined InSecTT requirement assigned to UCs and BBs

In more detail, all InSecTT requirements are collected in the SharePoint tool, where each requirement is described with the following 11 descriptive elements:

1. **ID:** An unique ID automatically given by SharePoint

2. **Short title:** Short and concise title of requirement in less than 32 characters
3. **Description:** Description of requirement in text form
4. **Rationale:** Rationale of requirement in text form that explains why is this requirement needed
5. **UCs:** Mapping to Use Case(s), i.e. a specification for which UC the requirement is important and needed. Following the described principles of InSecTT, one requirement can also be mapped to several UCs. All InSecTT UCs are offered as choice as shown in Figure 4.



<input checked="" type="checkbox"/>	T5.01 Wireless Platooning based on AI-enhanced 5G
<input type="checkbox"/>	T5.02 AI-enriched Wireless Avionics
<input type="checkbox"/>	T5.03 Wireless Security Testing Env. for smart IoT
<input type="checkbox"/>	T5.04 Intelligent wireless for port cross-domain appl.
<input type="checkbox"/>	T5.05 Connected solutions across health continuum
<input type="checkbox"/>	T5.06 Location awareness in healthcare
<input type="checkbox"/>	T5.07 Intelligent Transportation for Smart Cities
<input type="checkbox"/>	T5.08 Intelligent Autom. Services for Smart Transp.
<input type="checkbox"/>	T5.09 Cybersecurity in Manufacturing
<input type="checkbox"/>	T5.10 Robust resources management - large infrastr.
<input type="checkbox"/>	T5.11 Smart Airport
<input type="checkbox"/>	T5.12 Driver Monitoring and Distraction Detection
<input type="checkbox"/>	T5.13 Secure Industrial Communications System
<input type="checkbox"/>	T5.14 Secure and Resilient Collab. Manufacturing
<input type="checkbox"/>	T5.15 Safety/Security Public Transport Urban Env.
<input type="checkbox"/>	T5.16 Airport security - People Flow in Airports

Figure 4 UC choice

6. **BBs:** Mapping to Building Blocks (BBs), i.e. a specification which BB the requirement is mostly related to. This must be only one BB. In case a mapping to several BBs is desired, it is recommended to enter a dedicated requirement for each BB. The choices include all BBs as well as an additional choice for UC-specific adaptation (see Figure 5). This last choice can cover cases, where a direct mapping to BB is difficult and the requirement describes additional measures needed for the UCs as described in the principles above. However, this special choice shall only be used in exceptional cases.



Figure 5 BB choice

7. **Implementation release:** This is the project month for the planned implementation of the requirement. It can also contain several months in case of several iterations (see Figure 6). The option “Exceptional: Beyond InSecTT” shall be used for requirements that are not planned to be implemented in the life time of InSecTT but are still important to be considered, e.g. product requirement concerning serial production after InSecTT.

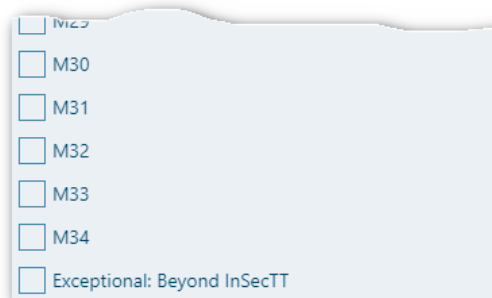


Figure 6 Implementation release choice

8. **Lead implementer:** Lead implementing partner as a choice field. In case there are other implementing partners, they can be specified in the comments field.
9. **Status:** The status of the requirement. Following choices are offered (see Figure 7):



Figure 7 Status choice

- a. Proposed: Initial status after entering requirement
- b. Confirmed in Iteration 1: Status after first requirements harmonization phase, this means the requirement is confirmed as ok and there are no other conflicting requirements.
- c. Merged: If a requirement is either redundant or overlapping with other requirements, it can be merged with another requirement in the harmonization phase. In that case, the status merged will be assigned to the requirement that will not be considered anymore for further work. The other remaining requirement will get the status of "Confirmed in Iteration X".
- d. Not implemented. This status is an exception for requirements that have a valid reason for not being implemented.
- e. Updated in iteration 2: This status follows an update of an already confirmed requirement in iteration 1.
- f. Confirmed and updated iteration 2/3: Similar as above

Note that the definition of status fields is subject to change during the first harmonization phase in InSecTT.

10. **Progress:** This field denotes the implementation progress of the requirement. After initially entering the requirement it is set to 0%. Afterwards it will be increased in the assessment phases. Details on this progress field will follow in InSecTT deliverable D1.5.
11. **Comments:** This field shall contain any additional information related to the requirement, e.g. explanation on new findings when implementing a requirement.

The requirements format with these fields has been defined in the first month of InSecTT within the Kick-off meeting as well as with online meetings in T1.2 considering all input of partners.

3.3 InSecTT requirements collection

The described requirements structure has been implemented in SharePoint in the beginning of InSecTT. Figure 8 shows instructions how to find the location within the SharePoint platform and to enter requirements.

A direct link to the requirements data base for InSecTT partners is:

<https://v2c2.sharepoint.com/sites/InSecTT/Lists/Requirements/AllItems.aspx>

Figure 9 shows how the individual requirement fields can be entered in SharePoint. The entry shall be intuitive for all project partners, therefore additional instructions are also displayed under each element.

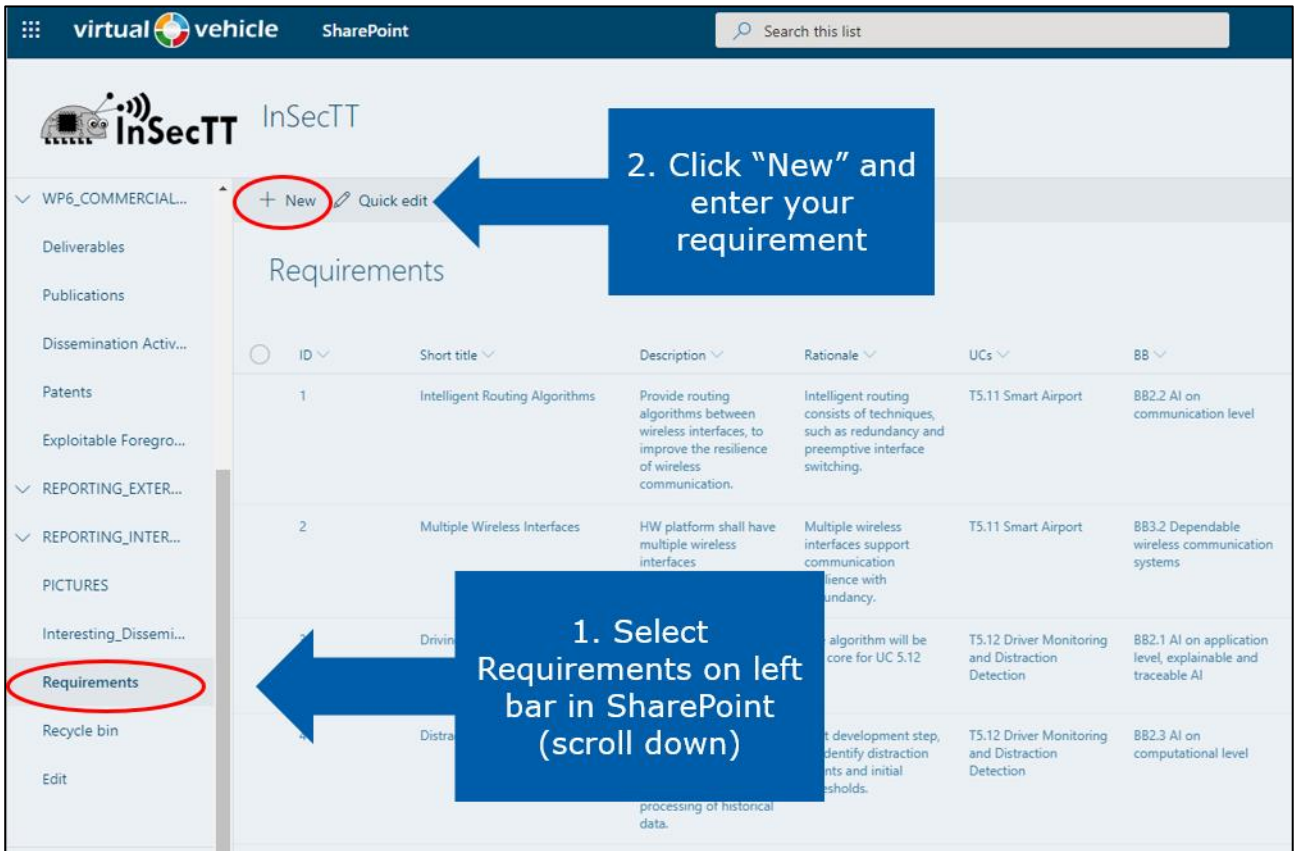


Figure 8 Requirements data base on the InSecTT SharePoint

The screenshot displays the InSecTT web application interface. On the left is a navigation sidebar with categories like Home, Documents, Project Calendar, and various work packages (WP1-WP6). The main area shows a table titled 'Requirements' with columns for ID and Short title. A 'New item' pop-up window is open on the right, containing a form with the following fields:

- Short title ***: Text input with placeholder 'Enter value here'. A red error message states: 'You can't leave this blank. Please give the requirement a short, concise title in less than 32 characters. Should not be the same as another, existing requirement.'
- Description ***: Text input with placeholder 'Enter value here'. A red error message states: 'You can't leave this blank.' Below the input is the label 'Description of requirement'.
- Rationale ***: Text input with placeholder 'Enter value here'. A red error message states: 'You can't leave this blank.' Below the input is the label 'Rationale - Describe why this requirement is needed'.
- UCs ***: Dropdown menu with the selected option 'T5.01 Wireless Platooning based on AI-enhanced 5G'. Below the dropdown is the text: 'Requirement must be related to a Use Case. Specify for which Use Case(s) the requirement is important and needed. One requirement can be applied to several UCs, if the requirement description matches to them. For a mapping to several UCs an alternative option would be to split the requirement into several requirements (e.g. when it is difficult to reflect this within a single requirements description).'
- BB ***: Dropdown menu with the selected option 'Select an option'. Below the dropdown is the text: 'Specify which BB the requirement is mostly related to. This must be one BB. For exceptional cases, if the requirement is very use-case related and cannot be mapped to a BB, you can choose "Exceptional: UC-specific adaptation". Please provide a reason for that in the description or comments field.'
- Implementation release ***: Dropdown menu with the selected option 'Select options'. Below the dropdown is the text: 'Month for planned implementation of requirement. Can also have several months in case of several iterations.'
- Lead implementer ***: Dropdown menu with the selected option 'Select an option'. Below the dropdown is the text: 'Lead implementing partner'.
- Status ***: Dropdown menu with the selected option 'Proposed'. Below the dropdown is the text: 'Initial status is "Proposed". Guidance on other values will follow at a later stage of InSecTT'.
- Progress ***: Dropdown menu with the selected option '0% - For initial entry 2020'. Below the dropdown is the text: 'Value used for assessment. 0% Initial entry of requirement / 40% Implemented on BB level to small extent / 70% Implemented on BB level to large extend / 100% Fully implemented on BB level'.
- Comments**: Text input with placeholder 'Enter value here'. Below the input is the text: 'Any other important information, e.g. explanation on delays or new findings when implementing requirement'.

At the bottom of the pop-up window are 'Save' and 'Cancel' buttons. A blue callout box with a white arrow points from the left towards the 'Short title' field, containing the text: '3. Enter all information in pop-up window'.

Figure 9 Pop-up menu for entering requirements

In order to target the overall goal of a manageable number of requirements, all partners are requested to fill in an appropriate number of requirements reflecting their planned work. An approximate value is to enter a range of 10 to 20 requirements per partner. Large and small partners in InSecTT may have deviations from those numbers.

The approach of requirements entry is a bottom-up approach. This means everybody fills in requirements, no matter if they overlap with other partners. Duplicate and similar requirements will be sorted out in the alignment phase that follows the requirements entry (see also following chapter).

3.4 Requirements timeline

The InSecTT requirements management approach and the related timelines do support a quick transition from the requirements entry and alignment stage into the execution and implementation phase.

InSecTT allows up to three implementation iterations, which means that from a requirements management perspective the biggest and most important entry and alignment activity will happen at project start, with two optional update cycles at the end of year 1 and year 2.

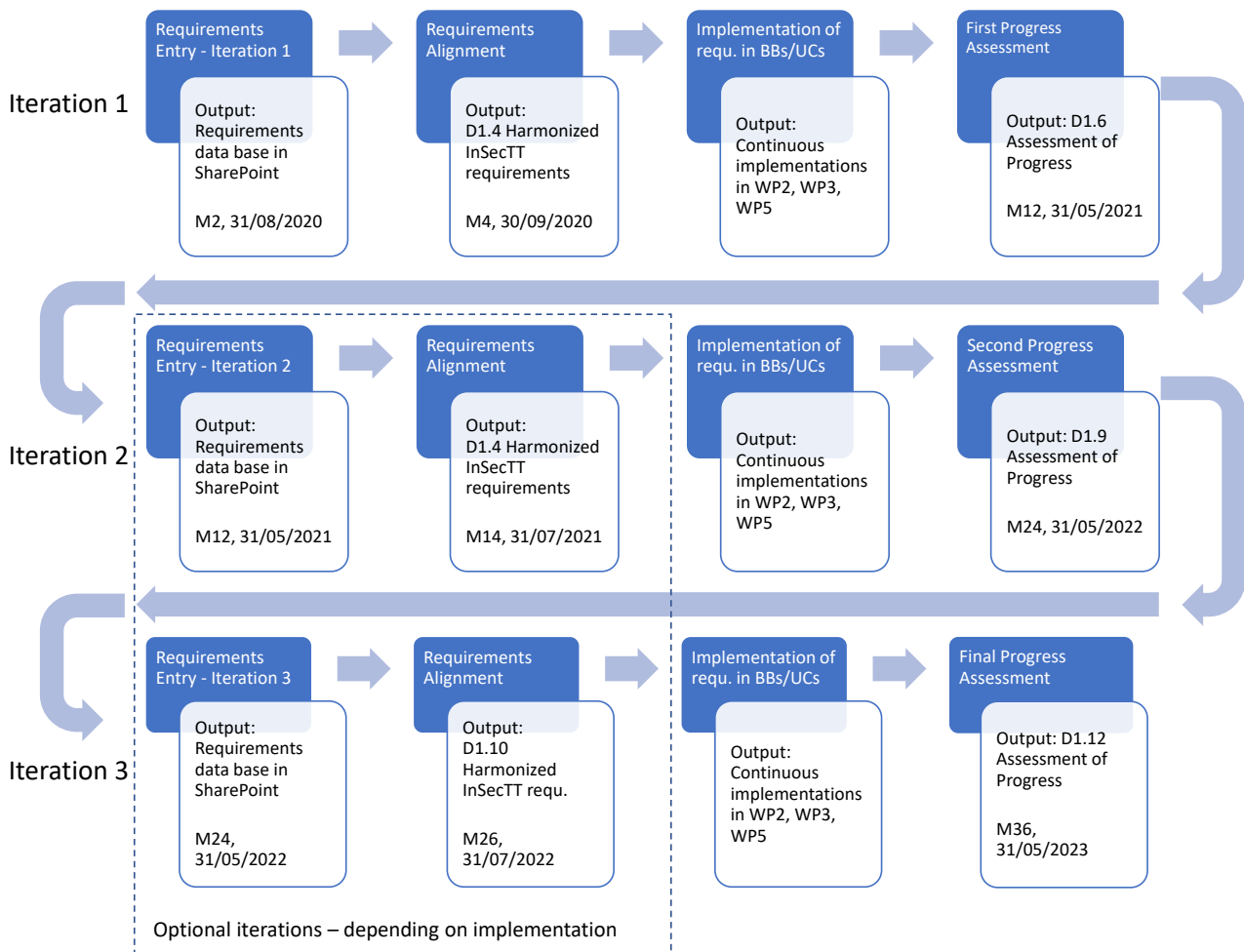


Figure 10 Timeline of InSecTT requirements management with 3 iterations

As shown in Figure 10, the initial requirements management iteration in InSecTT will be implemented with the following four concrete steps:

- Step 1: Requirements entry on SharePoint will be done by August 31, 2020
- Step 2: The alignment of requirements (e.g. grouping and elimination of duplicate requirements) will be done by September 30, 2020. Details on the method for alignment will be reported in the respective deliverable D1.4 [4].
- Step 3: Based on the requirements, implementation work will be carried out in various work packages.
- Step 4: At the end of the first project year, an assessment of technical progress will be performed based on the requirements. Details on the assessment methodology will be elaborated in a later stage and will be reported in the respective deliverable D1.6 [5].
- Afterwards, the given steps will be repeated in iteration 2 and 3. These iterations are optional and may be followed depending on the individual implementations. If only one iteration for the implementation is chosen, an assessment in three steps will still be performed to measure the progress over the project years.

3.5 Outlook to requirements alignment

The main idea of the requirements alignment or harmonization phase is to have a consistent set of requirements without duplicates and overlaps. In this phase all requirements will be individually discussed and analysed, if they can be confirmed for further project work. In order to clearly identify, which requirement has finished this step, the status field of the requirement will be used. All requirements that are kept directly or with changes will change their status to “Confirmed in Iteration 1”. If requirements are not kept, they will be set to status “Merged” or “Not implemented”. Details on this alignment process will be described in deliverable D1.4 [4].

3.6 Outlook to progress assessment

The main idea of progress assessment is to get quantitative indicators for the progress of InSecTT implementations. The progress of BBs, UCs and other implementations will be calculated based on the progress of individual requirements that are assigned to it. For that purpose the progress field in the requirement will be used. Details on the calculation method and the scale will be described in deliverable D1.6 [5].

4 DISSEMINATION, EXPLOITATION AND STANDARDISATION

This specific deliverable is in first instance an InSecTT internal guideline with the clear objective to support a fast and efficient requirements entry and alignment process, keeping in mind the ambition of re-use.

From this perspective this guideline is one instrument to support a successful exploitation of project results at a later point in time, by ensuring that Building Block requirements will meet Use Case needs and thus support competitive implementations and a fast time-to-market.

If the InSecTT requirements management and assessment approach proves to be the expected step-up from a methodological perspective, obviously project external dissemination will be considered on top.

5 CONCLUSIONS

The T1.2 core team that includes the InSecTT WP leads has developed this guidance document to facilitate an efficient requirements management process in InSecTT.

InSecTT principles are highlighted to provide the big picture, followed by a detailed description of the requirements entry process, as well as of related time-lines and the tool for the repository. An outlook is given to the optional iteration loops after year 1 and 2, and the required activities regarding assessment of progress and objectives.

We are convinced that this document will support InSecTT partners in running required activities in an efficient way and thus support ambitious timelines of the project, as well as overall targets in terms of re-use and interoperability.

The T1.2 team has already offered two sessions for efficient deployment of the InSecTT requirements management process and will make sure, that the comprehensive document will also be professionally deployed amongst all partners.

6 REFERENCES

- [1] InSecTT JU Grant Agreement incl. Description of Action (DoA), ECSEL Joint Undertaking, Grant Agreement No. 876038, Part B, signed by European Commission on 2020-05-20
- [2] DEWI – Dependable Embedded Wireless Infrastructure, European ARTEMIS project, see: <http://www.dewiproject.eu/>, March 2014 – April 2017
- [3] SCOTT – Secure Connected Trustable Things, ECSEL project, see <http://www.scottproject.eu>, Mai 2017 – October 2020
- [4] InSecTT Deliverable D1.4, Harmonized InSecTT Requirements (Iteration 1), due Sept. 30, 2020
- [5] InSecTT Deliverable D1.6, Assessment of Progress (Iteration 1), due May 31, 2021

A. ABBREVIATIONS AND DEFINITIONS

Term	Definition
AI	Artificial Intelligence
BB	Building Block
UC	Use Case