TRUSTS Trusted Secure Data Sharing Space

D2.4 Methodologies for the technological / business validation of use case results I

Document Summary Information

Grant Agreement No	871481	Acronym	TRUSTS
Full Title	TRUSTS Trusted Secure Data Sharing Space		
Start Date	01/01/2020	Duration	36 months
Project URL	https://trusts-data.eu/		
Deliverable	D2.4 Methodologies for the technological/business validation of use case results I		
Work Package	WP2		
Contractual due date	30/06/2020	Actual submission date	26/06/2020
Nature	Report	Dissemination Level	Public
Lead Beneficiary	EBOS		
Responsible Author	Aspa Filippou		
Contributions from	FNET, FORTH, REL, TUD, RSA, LST		

Revision history (including peer reviewing & quality control)

Version	Issue Date	% Complete	Changes	Contributor(s)
v0.1	07/02/2020	5	Initial Deliverable Structure	Aspa Filippou (EBOS)
v0.2	31/03/2020	40%	1st Draft	Aspa Filippou (EBOS) Ioannis Markopoulos (FNET)
v0.3	30/04/2020	55%	2nd Draft	Aspa Filippou (EBOS) Ioannis Markopoulos (FNET)
v0.4-v0.8	26/05/2020	85%	3rd Draft	Christos Skoufis (EBOS) Ana Maria Florea (REL) Aspa Filippou (EBOS)
v0.9	01/06/2020	85%	Document review	Ioannis Markopoulos (FNET) Mark de Reuver (TUD) Stefan Gindl (RSA) Ana Maria Florea (REL)
v1.0	05/06/2020	100%	Final version sent for Peer and Consortium Review	Christos Skoufis (EBOS) Aspa Filippou (EBOS)
v1.1	15/06/2020	100%	Peer and Consortium Review results	Ioannis Markopoulos (FNET) Evangelos Kotsifakos (LST) Rosa Araujo (LST) George Margetis (FORTH)
V1.2	25/06/2020	100%	Final version for submission	Aspa Filippou (EBOS)

Disclaimer

Copyright message

Table of Contents

List of Figures

List of Tables

Glossary of terms and abbreviations used

Abbreviation / Term	Description
AML	Anti-Money Laundering
ACT	Automated Confirmation Transaction
API	Application Programming Interface
DoA	Description of Action
GA	Grant Agreement
KPI	Key Performance Indicator
KYC	Know Your Customer
PDCA	Plan–do–check–act
QoE	Quality of Experience
QoS	Quality of Service
SAR	Suspicious Activity Report
SEO	Search Engine Optimization
SLA	Server Level Agreements
SUS	System Usability Scale
TDD	Test-Driven Development
TRUSTS	Trusted Secured Data Sharing Space
TUD	Delft University of Technology
UAT	User Acceptance Test
UC	Use Case
WP	Work Package

1 Executive Summary

This is the first version of the Project's D2.4 Deliverable "Methodologies for the technological/business validation of use case results I" addressing the Task 2.3, along with the work that has been performed under this Task. The deliverable D2.4 is part of the WP2 of the project "TRUSTS Trusted Secure Data Sharing Space". The deliverable starts with an overview of defining the methodologies for the technological and business validation of the TRUSTS platform within and across each vertical use case. In addition, it includes the definition of test reports format and benchmarking for the validation of the KPIs.

In the following chapters of this deliverable, a description is given with regards to the procedure of the Test-Driven Development (TDD) Methodology and toolset for the analysis of the data marketplace technologies and the vertical Use Cases (UC) for the purposes of the technological validation that will be held during the life period of the project. Additionally, a description is made for the Lean Start-Up Methodology with respect to the business validation during the duration of the project in order to receive end-users feedback and to set the metric and parameters and also to present the KPI's validation so as to enable us to focus on the lessons learned and to conclude for the next actions on a project level.

Furthermore, there is a focus on the methodology to be followed during the use case trials and what needs to be tested, which will enable the validation and evaluation of the functionality and performance of the data marketplace in order to deliver outputs that have commercial value and potential. Hence there will be given a significant emphasis on:
- Quality of Service (QoS) to give a qualitative measurement of test execution,
- Quality of Experience (QoE) to gauge objective data marketplace user experience.

With respect to the TDD, an incorporation of unit tests / user acceptance tests (UAT), for technological validation of the project, will be gathered. As per the business validation a number of business validation templates will be used for the evaluation of TRUSTS and the offered services as a whole. Also, the methodology will define how the interaction with the end-users could be achieved taking into consideration the industry-specific functional specifications appropriate for an EU and worldwide data marketplace, as part of the business validation process.

Finally, this deliverable presents the System Usability Scale (SUS) methodology as a separate process for the overall evaluation of TRUSTS data marketplace and its portfolio as part of commercial value testing and users experience based on QoS and QoE.

Another version with respect to T2.3 regarding the methodologies for the technological/business validation of UC results will be provided with D2.5 in M24.

2 Introduction

The successful creation and adoption of a pan-European data sharing space will mark a milestone in the growth of the new data economy. Emerging Data Ecosystems that enable large-scale data to be securely connected, valorized and shared, rely on the Europe's purposes under the scale of Horizon 2020 and the technological development. This is why TRUSTS 'Trusted Secure Data Sharing Space' will make the difference in the sphere of data technology and data innovation while data market is empowering in Europe and worldwide.

TRUSTS, supporting the emergence of a European data market and economy based on secured, safe and GDPR–compliant data exchanges aim to develop a data-sharing platform for secure, trustworthy and compliant data exchanges to GDPR regulation.

In this specific deliverable, the scope is to define the methodology and toolset for a comprehensive and robust analysis of the data marketplace technologies and the vertical use cases that will be implemented within the framework of the TRUSTS Project.

The major requirement here is for the project to deliver outputs that have commercial value and potential. Hence there will be a significant emphasis on the Quality of Service (QoS), aiming at the overall platform performance evaluation and not only to its network oriented features, and the Quality of experience (QoE) along with an incorporation of Test-Driven Development (TDD) methodology including unit tests and user acceptance tests, for both technological and business validation.

TDD methodology will also define how the interaction with the vertical end-users will be achieved, taking into consideration the specifics of T2.1. The inputs will include apart from the business case itself, an end-user feedback from their direct engagement in the trials of the vertical use cases.

A detailed set of metric parameters considered for the business validation of each UC will be developed, which will be interrogated and quantified as part of the business validation process with the end-users. Moreover, threshold limits of the results will be defined per target KPI (based on the requirements of each vertical use case).

The corresponding outputs will be validations that will allow to identify the use cases that have the highest commercialization potential in order to progress to the next step of creating a data marketplace service portfolio. In order to identify this, we will use a set of questionnaires, surveys and focused group workshops directly engaging the industrial associates of the consortium partners.

2.1 Mapping TRUSTS Outputs

The purpose of this section, is the mapping of TRUSTS Grant Agreement commitments, both within the formal Deliverable and Task description, in relation to the project's respective outputs and work performed.

Table 1: Adherence to TRUSTS GA Deliverable & Tasks Descriptions

TRUSTS Task		Respective Document Chapter(s)	Justification
T2.3 Testing framework and benchmarking	This task will focus on defining the methodology and toolset for a comprehensive and robust analysis of the data marketplace technologies and the vertical use cases being created within the TRUSTS project. Working closely with the partners defining the scenarios to be trailed in the TRUSTS environment, we will specify formats for a suite of test cases to measure the functionality and performance of the innovative solutions being put forward. These test artefacts will be captured in a suitable test case management tool that will integrate seamlessly with the development process. Based on the requirement to deliver outputs that have commercial value and potential, there will be a significant emphasis put on Quality of Service (QoS) to give a qualitative measurement of test execution and on Quality of Experience (QoE) to gauge objective data marketplace user experience. The test process will fit neatly into the project's iterative agile development process and allow for implementation of a Test-Driven Development (TDD) methodology incorporating unit tests and acceptance tests. The methodology will entail acceptance test procedures for conducting both the technological and business validation of the use cases considering the associated service management. Threshold limits for the benchmarking of the results will also be defined per target KPI based on the requirements stemming from each vertical use case. The methodology will also define how the interaction with the vertical end-users will be achieved taking into consideration the specifics of T2.1. For the business validation, partners will	Sections 3-5	*Section 3* • *Test case validation toolsets.* • *Methodologies Processes and Dependencies in respect to the Business and Technological Validation.* • *Agile testing and validation process life cycle.* *Section 4* • *TRUSTS data marketplace business validation* • *Lean Startup Methodology* • *Business validation templates.* • *Business validation and KPI's* *Section 5* • *TRUSTS data marketplace technological validation* • *Test Driven Development Methodology* • *Testing and acceptance tests* • *Templates to be utilized* • *Technological Validation and interaction with WP4* *Section 6* • *QoE and QoS validation* • *SUS Validation Methodology* • *TRUSTS validation via*

	use the lean start-up methodology that centres around on the main motivations of a business. The inputs will include apart from the business case itself, end-user feedback from their direct engagement in the trials of the vertical use cases. The corresponding outputs will be validations that will allow to identify the use cases that have the highest commercialisation potential to progress to the next step of creating a data marketplace service portfolio. We will use a set of questionnaires, surveys and focused group workshops directly engaging also the industrial associates of the consortium partners. A detailed set of metrics parameters considered for the business validation of each UC will be developed, such as those listed in section 1.3 for each use case, which will be interrogated and quantified as part of the business validation process with the end-users.		*QoE, QoS and SUS*
TRUSTS Deliverable			
D2.4 Methodologies for the technological/business validation of use case results I. This is the first version of the two reports defining the methodologies for the technological and business validation of the TRUSTS platform within and across each vertical use case. It also includes the definition of test reports format and benchmarking for the validation of the KPIs.			

2.2 Deliverable Overview and Report Structure

The following section provides an overview of the Deliverable's structure as well as a detailed description of the plan of action in compliance with the expected outcomes of the T2.3.

A special attention is given on the key elements of a well-balanced methodologies for the technological and business validation of the TRUSTS platform of use case results along with the end-user feedback, which are the metric and parameters and as well as the KPI's validation and threshold limits.

A summary of the sections of this report is included below.

- **Section 3,** presents the test case validation toolsets. Describes and analyzes the Methodologies Processes and Dependencies with respect to the Business Validation and Technological Validation along with the testing and validation process life cycle based on agile way of working.
- **Section 4**, gives information on the TRUSTS data marketplace business validation using Lean Startup Methodology. Furthermore, business validation templates are provided and there are also presented the KPI's as a business validation method for the 3-business oriented use-cases.

- **Section 5**, gives information on the TRUSTS data marketplace technological validation based on Test Driven Development Methodology with focus on testing and user acceptance tests along with templates to be utilized.
- **Section 6,** presents the Quality of Experience and the Quality of Service for the purposes of the measurement of test executions. Commercial value testing and user's experience are based on SUS Validation Methodology in this section.

3 Methodologies and Dependencies

This section describes in more details the TRUSTS validation methodologies along and dependencies. This addresses the objectives of T2.3 as per DoA related to project Objective 1[1], on defining the methodology and toolset for a comprehensive and robust analysis of the data marketplace technologies and the vertical use cases (UC's) being created within the TRUSTS project life cycle.

Thus, this section introduces the production and assessment of the methodologies for the testing, validation and benchmarking of the results as well as for the technological and business validation of the UCs through an agile-based iterative process approach.

3.1 Test Case Validation Toolset

This section defines the toolset that will be used for the test case validation. The diagram below presents the architecture of this validation toolset, including both business and technological validation along with the commercial value testing and user's experience for TRUSTS data marketplace.

As presented, the business validation will be performed by utilizing the "Lean Startup[2]" methodology. Business validation will be implemented throughout business validation templates, functional requirements templates and business questionnaire template as descripted in section 4.1 and also via KPI's validation as presented in section 4.2.

Furthermore, the technological validation will be performed by utilizing the "Test-Driven Development" (TDD) methodology via unit tests and/or user acceptance tests, as presented in section 5.2 and 5.2.1, in order for the end-users to perform application testing and provide test results along with their feedback. A tight set of unit-tests in combination with TDD allows for rapid development cycles and facilities CI/CD[3].

Additionally, TRUSTS validation as part of commercial value and user's experience will be held based on the System Usability Scale (SUS) scoring methodology as mentioned in section 6.3. SUS Methodology will enable the overall scoring of TRUSTS marketplace and the offered services based on Quality of Experience (QoE) and Quality of Service (QoS) as presented in section 6.4.

---

[1] DoA Objective 1: To analyse the EU & worldwide challenges and trends for data-sharing and define the requirements for the provision of a multi, concurrent and cross-domain, secure and scalable end-to-end (E2E) data marketplace service.

[2] http://theleanstartup.com/principles

[3] CI/CD is a method to frequently deliver apps to customers by introducing automation into the stages of app development. The main concepts attributed to CI/CD are continuous integration, continuous delivery, and continuous deployment. CI/CD is a solution to the problems integrating new code can cause for development and operations teams. (https://www.redhat.com/en/topics/devops/what-is-ci-cd)
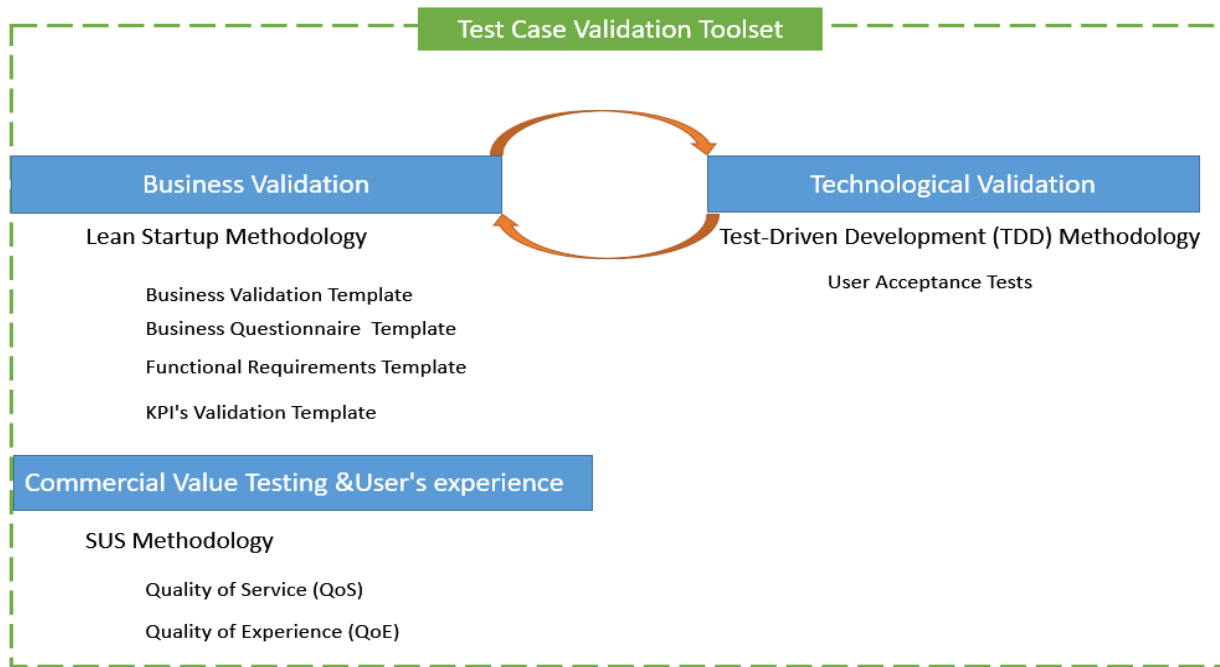
Figure 1: Validation Toolset Diagram

This is an analytic introduction information about how the validation aims to be performed over the project lifecycle along with the toolset that will be utilized. All the above-mentioned methodologies and validation activities along with the agile way of working are analysed in detail in the following sections including the relevant templates that enable those validations for the Consortium.

Concluding, the Validation Toolset is basically a set of methodologies to be applied and templates to be used by the partners of the Consortium in order to validate the progress on the implementation, the data marketplace services and UCs and the overall alignment with the projects' outputs and objectives. In order to design those templates, a set of validation methods and procedures combined with validation methodologies have been adopted, as presented in the Figure 1.

3.2   Business Validation and Technological Validation Dependencies

This section presents how technological validation and business validation go hand in hand allowing us to capture project objectives and satisfy end-user needs via the overall data marketplace evaluation over the three defined UCs. In TRUSTS the vital vision is that end users will be in place to perform progressive technological and business validation of greatly innovative and future oriented UC's in near real-time. There will be an execution of technological and business validation relying on agile methods to cater for the needs of dynamic development.

Figure 2 focuses on the time plan (timing and the duration) of business and technological validations (as part of the toolset), in line with the project plan and the defined projects' milestones. Moreover, as presented in the Figure 2, following an agile methodology (analyzed in Section 3.3), 3 sets of business validation and 2 sets of technological validations have been identified, allowing the interaction between the business needs, business models and the technological enablers, over the projects' lifecycle.

Validation Type	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18
	Jan-20	Feb-20	Mar-20	Apr-20	May-20	Jun-20	Jul-20	Aug-20	Sep-20	Oct-20	Nov-20	Dec-20	Jan-21	Feb-21	Mar-21	Apr-21	May-21	Jun-21
						MS1						MS2						MS3
1st Business Validation																		

Validation Type	M19	M20	M21	M22	M23	M24	M25	M26	M27	M28	M29	M30	M31	M32	M33	M34	M35	M36
	Jul-21	Aug-21	Oct-21	Nov-21	Dec-21	Jan-22	Feb-22	Mar-22	Apr-22	May-22	Jun-22	Jul-22	Aug-22	Sep-22	Oct-22	Nov-22	Dec-22	Jan-23
					MS4													MS5
1st Technological Validation																		
2nd Business Validation																		
2nd Technological Validation																		
3rd Business Validation																		

Figure 2: Toolset and Validation Time Plan

More details on what it is validated and how this validation is performed along with the involved partners and templates to be used, is presented below.

- **1ˢᵗ Business Validation**: To be performed from M7 to M10 by the UC participants, allowing the validation of the functional requirements as those are listed in D2.2 (using the Table 3 "Functional Requirements Validation Template"), and business information as those will be collected (using the Table 2 "Business Validation Template" and Table 4 "Business Validation Questionnaire Template"). This validation is in align with the Milestones timeline since it is initiated right after Milestone 1 "Project setup" (M6) and ends before Milestone 2 "End of first period" (M12). This will enable the business modelling under WP7 (in collaboration with T7.1 Sustainable Business Models) and support the technological engagement since the validation output will be an input back to WP2 (T2.2 "requirements" and T2.4 "architecture"), but also to WP3 ("platform implementation"), and WP4 (T4.1, T4.2 and T4.3 "Privacy preserving and data anonymization").

- **1ˢᵗ Technological Validation**: To be performed from M19 to M22, allowing the validation of the architectural framework and technical specifications (T2.4) along with the work under the T3.5 ("Initial Platform and integration"). In addition, with respect to the technological validation we have to evaluate the respective processes (as described in D2.2) as well as the rest of the components implemented under the WP3, such as smart contracting. This validation will give a feedback to T2.4 and WP3, and will be in align with Milestones timeline since it is initiated right after Milestone 3 "First Pilot Deployment" (M18). This validation will be performed by the UC participants during the 1st trial period, allowing them to check and validate the outcome of the technical implementation through predefined scenarios and document the results using the User Acceptance Test (UATs) Template, presented in Table 7.

- **2ⁿᵈ Business Validation**: Right after the completion of the 1ˢᵗ Technological Validation and following the iterated process between the technological and business validations, from M23 to M24 the UC participants should again re-evaluate their needs from the business perspective

that might be slightly changed or enhanced in a year from their 1st input (1st Business Validation).

For this re-evaluation, the templates completed during the 1st Business Validation will be enhanced or modified (if needed based on UC participants business needs). This validation period will also allow the evaluation of business models implemented under T7.1. This timeline is in align with Milestones timeline since it is initiated after Milestone 3 "First Pilot Deployment" (M18), it is during the period of the 1st UCs Trials, and ends on Milestone 4 "End of second period" (M24).

- **2nd Technological Validation**: To be performed from M25 to M29, allowing the validation of the Marketplace and the provided services during the 2nd set of UC trials by utilizing the defined test procedures and the reporting structure, and validation of results in regards to technology. This last round of technological validation will also evaluate the complete environment from a technical, performance, expandability (e.g. federation), etc. point of view and define the quality of the implementation. The output here can be an input back to WP3 and WP4 for the refinement of the implemented solution (marketplace). This validation will be in align with Milestones timeline since it is initiated right after Milestone 4 "End of second period" (M24) and performed by the UC participants during the 2nd set of UCs trial period, allowing them to check and validate the outcome of the technical implementation through predefined scenarios and document the results using the User Acceptance Test (UATs) Template as per Table 7.

- **3rd Business Validation**: To be performed from M30 to M33, allowing the evaluation of the complete environment from a performance and business point of view, via the measurement of KPIs and validation of their results in order to define the gap towards commercializing the environment. This last round of business validation will be performed again by the UC participants by utilizing the KPIs evaluation templates in Tables 4, 5 and 6. The output of this final business validation shall be an input to WP7.

The above validation plan is in align with the project where the implementation and testing plan for the pilots, should be ready by M14 for the first demonstration phase and updated by M25 for the second demonstration phase. The first phase of use case trials should be completed by M24 and second phase by M32.

Several methodologies, such as agile iterative process, agile approach, design thinking, agile methodology, and co-creation, have been referred in the proposal as the most promising approach. However, based on DoA, the Deming's Plan-Do-Check-Act[4] will act as the fundament for UC's iterations. These aforementioned approaches, even if they have some differences, also have several components which are essential for the implementation of TRUSTS.

a. The specific use case scenario will be tested and technically validated in several iterations, continuously refining the technological implementation of the scenario.
b. It is also a mutual consideration that it is better to test as much as possible in early phases, in order to avoid investments which are not addressing to a user's need in the market.
c. A clear understanding of the stakeholders' needs and what is valuable and viable for the data marketplace.
d. To understand that the problem or addressed need is clearly understood at some point of time, and that the iterations are concentrating on the technical adaptation and validation.

---

[4] https://www.mindtools.com/pages/article/newPPM_89.htm

3.3 Testing and Validation Process Life-Cycle and Agile Way of Working

The entire testing and validation process, along with the concept as described previously, is taking into consideration the agile methodology on software development to be followed as a general way of working over the project lifecycle.

Agile methodology is often compared with the waterfall model (mostly in the development period); however, agile approach is generally considered to be better since it uses an incremental approach where a sample prototype is discussed with the end-user. The prototype helps to understand the key aspects, including the requirements and the consecutive prototypes reflect the changes done in the previous prototypes. This keeps happening until the end-user is satisfied, providing better end-product. The idea is to maintain product's quality in the entire phase of development. While comparing the waterfall model and the agile methodology (Figure above), we concluded that:

- Each phase is a result of the process of previous steps
- The process should be repeatedly checked for consistency
- A single iteration would not give a clear picture of the process.

Moreover, following are a number of reasons why an agile methodology has been chosen to be applied in general, during TRUSTS lifecycle:

- It is more flexible, fast, lean, responsive, and consistent.
- It focuses on end-user and is more communication-oriented
- It is more flexible by adapting to the change of requirements and end user needs.
- It follows best practices, such as Iterative Development etc. that help in getting high-quality software very quickly.

With respect to the testing and validation process life -cycle, based on agile way of working these above-mentioned components or phases illustrated as per Figure 4, demonstrating PDCA procedure that will be used for TRUSTS.

An integration of design thinking as part of the methods along with a stage of stabilized value problem scenario and personas[5] will be used. Personas are fictional characters, representing the different end-users in a consumer setting or for instance ecosystem actors/stakeholders in a professional setting. Such imaginary personas are imagined to develop or use your service, product or platform in different ways. The creation of personas enables a better understanding of pain points (problems), needs, experiences, value perceived, behaviors and goals when developing and testing new services/products or processes.

PDCA (plan–do–check–act, also known as the Deming circle (Figure 4), is an iterative four-step management method used in business for the control and continuous improvement of processes and products. The PDCA procedure's steps are listed below:

- **Plan:** Recognize an opportunity and plan a change.
- **Do:** Test the change. Carry out a small-scale study.
- **Check:** Review the test, analyze the results, and identify learnings.
- **Act:** Take action based on what you learned in the study step. If the change did not work, go through the cycle again with a different plan. If you were successful, incorporate what you learned from the test into wider changes. Use what you learned to plan new improvements, beginning the cycle again.

---

[5] https://www.alexandercowan.com/venture-design/
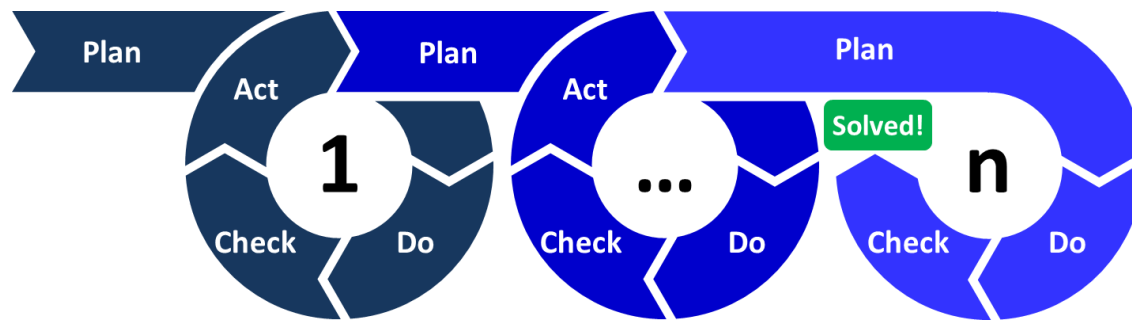
Figure 3: PDCA Procedure / Deming's cycle[6]

Taking into account the above, we define that technological and business validation refer to two different aspects of working agile. There is an interdependency between business validation and technological validation as this is previously mentioned in Section 3.2. Throughout the technological validation we test if it is possible to solve the pain point technology wise and also to measure it. Along with the establishment of the real pain existence, we need to define the business value of solving this pain. Following technical validation, the business validation process anticipates the confirmed business value on the level of a specific UC scenario, to a market level.

Figure 5, presents the relationship between business and technological validation. Iterative feedback loops are included so as to illustrate if a test can fail to meet the validation requirements in order to go back to the beginning of the whole process.
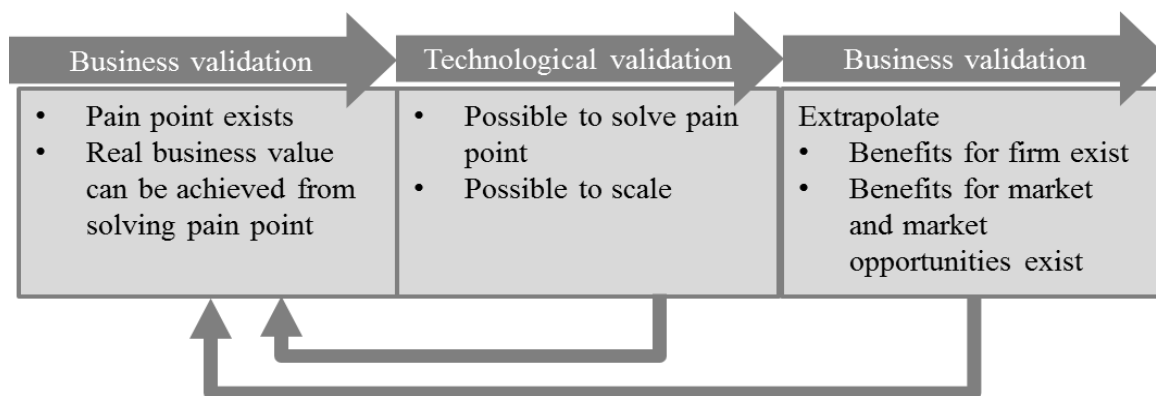

Figure 4: Illustration of relationship between business and technological validation

If the initial business validation parameters are robust, a technological validation process is carried out. This focuses mainly on whether the problems/pain points can be solved using TRUSTS. For instance, will the service required improve the stakeholder/user experience? Will the reliability, coverage and accuracy be improved as compared to current situations in data marketplaces?

If the technological validation parameters also are confirmed we continue to a next step of business validation. This is where we go beyond the single UC, and document on an aggregated level the size of the business value and opportunities for the data marketplace. To conclude, if markets cannot be confirmed, we return to step 1 or 2.

---

[6] https://asq.org/quality-resources/pdca-cycle

4 Business Validation

This section describes the methodology for the business validation in TRUSTS. The objective is to validate the three UC's business wise and develop business plans for the UC's with the highest commercial potential. This business validation methodology is grounded in agile, design thinking and lean start-up methodologies and supports the methodology presented in TRUSTS work plan, where lean start and agile approaches are advocated.

Business validation is a must in view of the essential risks related with entering the market without knowing if you have got it right since the beginning. There are various steps associated in validating a business that entrepreneurs must follow. This is important as there is always a chance to amend the business idea and taking the right approach, in case there is any delay during validation.

In TRUSTS work plan, the business validation methodology is based on agile, architecture thinking and lean start-up methodologies where the latter supports the methodology presented in DoA, and centers around the main motivations of a business.

Thus, for the business validation, partners will use the lean startup methodology which focuses on major impulses of a business. The inputs will include apart from the business case itself, the end-user feedback from their direct engagement in the trials of the UC's. The corresponding outputs will be validations that will allow to identify the UC's that have the highest commercialization potential so as to progress to the next step of creating a data marketplace service portfolio.

Moreover, part of the business validation is also the validation of the KPIs, as those were defined in DoA, specific for each UC and are presented in section 4.2 below.

4.1 Lean Startup Methodology

Lean start-up is a methodology aiming to develop businesses and products in order to compress the various periods of product development as well as to disclose if an upcoming proposed business model could be characterised as feasible. This methodology also favours the experimentation (over an elaboration of the overall planning), customer feedback (over intuition), and iterative design (over traditional "big design up front" development) .It is a methodology based on "validated learning", that is, to validate the hypotheses little by little before having the final product and start to scale the business. Shorten development cycles defined, launching different proposals for a time period and obtaining valuable feedback from the end-users, with which to improve the next final version of the product.

There are 5 fundamental steps in the lean start up methodology as described below:

1. **Pose a hypothesis:** Part of a problem to solve and explain why they would be willing to pay for your offer. To identify the problem / pain, we can conduct a series of interviews with our potential clients and identify what really concerns them. We must know if the problem is painful enough to attack it.
2. **Validate the hypothesis:** From creating a product or service with the basic minimum characteristics to verify if it is what the market wants until a demonstration of how it works, everything is possible. The objective is to know if people would want it and buy it. This first validation will be of the "early adopters"[7], the first users who will use it and the ones most susceptible to try new things in our sector.
3. **Measure the hypothesis:** The best way to know what metrics you will implement is to identify what are the steps to follow to get to your offer and how many times they turned to them to

---

[7] In startup language they are often called the three Fs: "family, friends, and fools"

buy. It is essential to identify the KPIs of a product and to measure them, in order to find out if we meet its objectives and direct us on perfecting our product.

4. **It generates validated learning:** It means that along the way adjustments and changes have been made in the product or service. It is essential to listen to all stakeholders (people directly or indirectly involved in the product / service) and incorporate their feedback.

5. **Repetitive cycle:** You start up the previous steps once again with an improved product or service and start over.
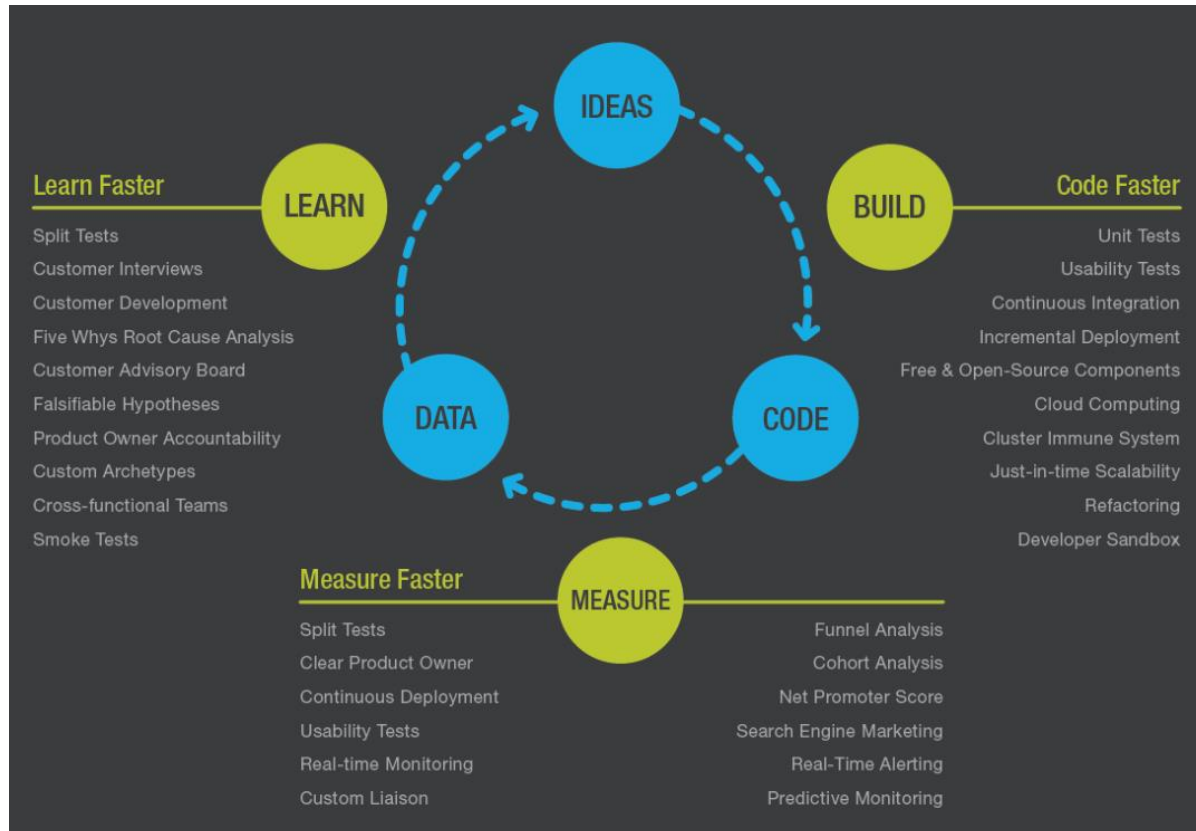
Figure 5: Lean start up – agile ways of work[8]

4.1.1 TRUSTS Business Validation via Lean Startup

In TRUSTS we aim that end users will be able to perform business validation of inventive and future oriented UC's of a data marketplace and to initiate that, a lean startup methodology aims to be applied. The core flow of the Lean Startup can be seen in the Lean Cycle in Figure 6 showing a set of steps used to take a lean approach in regards to the validation of TRUSTS data marketplace.

In TRUSTS, partners will use the lean start up methodology in respect to the business validation process, that focuses around the main motivations of a business. The inputs will include apart from the business case itself, end-user feedback from their direct engagement in the trials of the vertical UC's. A detailed set of metrics parameters considered for the business validation of each UC will be developed and will be interrogated and quantified as part of the business validation process with the end-users.

---

[8] http://www.rodrigocordero.org/en/how-to-apply-the-lean-startup-methodology/

Figure 7 presents the process of business validation opportunities withing the TRUSTS UC's, which is based on agile design thinking and on lean start up methodology.
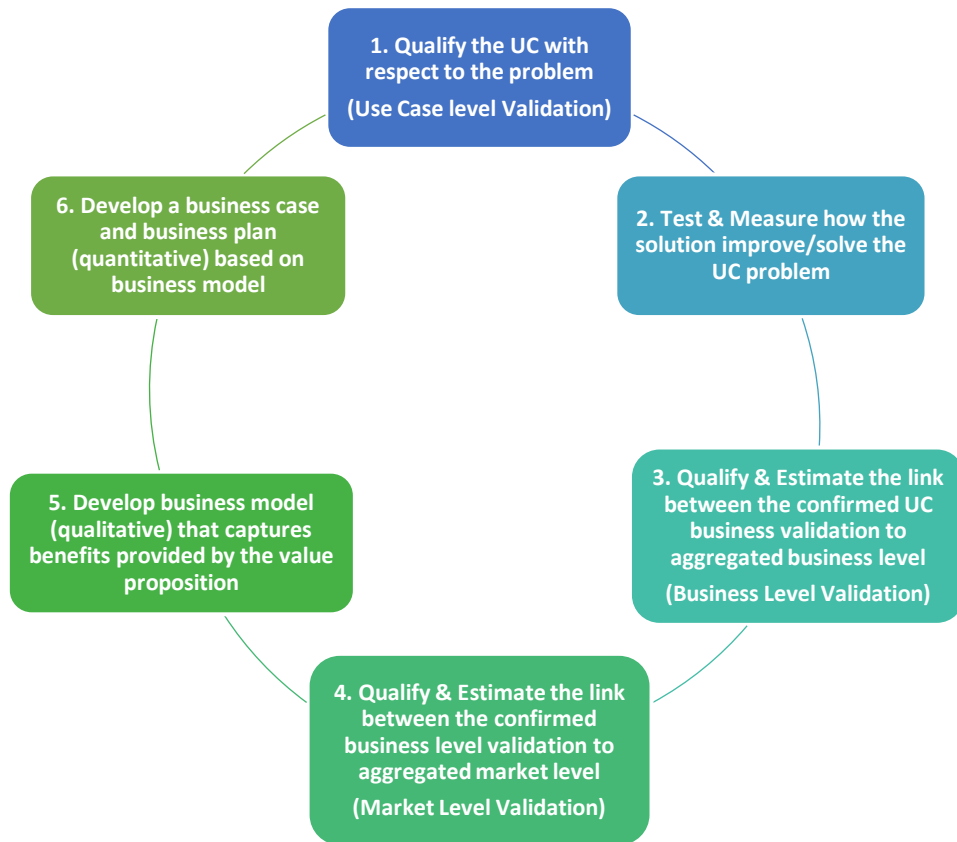
Figure 6: Stepwise Methodology for TRUSTS business validation

The above method is linear meaning that each element will provide input to the following steps, although it also includes repetitions between the steps. That is, value propositions referring to the solutions to problems described and tested in step 1 and step 2 should be included in the business model framework as per step 5 and it could also imply to return to step 2 again in respect to the business validation.

TRUSTS project has a goal to create viable business plans for its defined targets. The creation of those business cases and business plans will be strongly informed by the business outcomes of the three UCs which will all provide unique input to the final business plans.

Thus, based on Lean Startup Methodology business validation templates have been defined and provided in order to verify the business needs of UC's and those of the end users as well. The objective in this step is to build a UC that is to be validated business wise following the validation process life-cycle and agile way of working. In general, interviews with UC "owners" and their contacts, and experts from within and outside their own firm, will provide the primary sources of information. The Business Validation template as per Table 2 will help to deliver an initial level of consistency across each of the UC's and provide a solid foundation on which to build further business analysis.

It is anticipated that the information gathered from each UC will require several iterations (e.g. via calls, face-to-face meetings/works shops and survey data from online questionnaires etc.) to gather sufficient level of detail that will provide the necessary input for the creation of Business Models in WP7, T7.1 'Sustainable Business Models'. WP7 is about business model for the data marketplace, where WP2 is

about business model for the use cases that utilize the data marketplace. So, those two are highly related due to the fact that the platform business value depends on what the users of the platform defines as their business value.

Following the Lean Startup Methodology, and the functional requirements as per D2.2 "Industry specific requirements analysis, definition of the vertical E2E data marketplace functionality and use cases definition I", we have proceeded in preparing the aforementioned templates, taking into consideration the UC's itself and the demands of the end users. This approach is systematic, and this will ensure that the analysis is specific for TRUSTS, data marketplace, as well as the UC contexts.

The primary source of information for completing this template will be derived from the UC owners. The Business Validation template described in Table 2 will help to deliver an initial level of consistency across each of the UCs and provide a solid foundation on which to build further business analysis.

The table will be filled during the 1st Business Validation period (M7 to M10) prior to the execution of the UC's so as to define the background, the personas and the problem as well as after the implementation of the UC's in order to provide the expected/potential benefits arising from the UC's. This situation of 'to be' should demonstrate the objective 4, to present the added value of the TRUSTS Platform in three business-oriented UC's which showcase the sharing, trading, (re)use of data and services and result in added value generated through innovative applications built on multiple open and proprietary data sources.

Table 2 presents the template to be followed and filled by UC participants for each use case accordingly for the business validation. This template will be used in the 1st Business Validation (M7-M10) for the collection of the business information so as to define the needs from the business perspective of UC participants. Furthermore, this template completed during the 1st Business Validation will be enhanced or modified (if needed based on UC participants business needs) for the 2nd Business Validation (M24-M25) so as to re-evaluate their needs from the business point of view. This template is focusing on the areas listed below:

- **Background:** *textual description of the business process and context surrounding the UC.*
- **Personas:** *describe ALL actors/users who are directly impacted by the UC.*
- **Problem:** *Describe in detail the problems that* each persona*/stakeholder currently experience (AS-IS today before TRUSTS).*
- **Expected Benefit:** *the benefit that each persona hopes to achieve from the UC (after TRUSTS is implemented).*

Table 2: Business Validation Template

Background
Please provide a textual description of the business process and context surrounding the UC.
What is the general context of the UC? (describe the Organisation / business situation)   *Under what circumstances does the UC arise?*   *How often?*   *Other information?*
Describe the Personas   *Please describe ALL personas who are **directly** impacted by the UC*   *Describe each persona of the TRUSTS (Consumer? Org/Business operations? Technology? Etc.);*   *Please be as specific and detailed as possible about exactly what each persona does.*
*Describe the **end user personas** (e.g. different types of consumers; operators in a data marketplace?)*

Persona Name	Persona Role

*Describe the **application provider(s)** (who builds and supports the application?)*	
Persona Name	*Persona Role*
*Describe **other actors** directly involve/impacted by the UC?*	
Persona Name	*Persona Role*
End user personas	

Describe the Problem

Describe in detail the problems that each persona*/stakeholder currently experience (AS-IS today before TRUSTS)*

Personas (who exactly?) experience this **problem** (what exactly?) when doing this **task** (when does it occur?)   OR

Personas (who exactly?) experience this **problem** (what exactly?) because of this **constraint** or limitation (when does it occur?)

End user Persona	
Problem	
Task / Constraint	
How is it addressed now? (Pre- TRUSTS)	
Application Provider Persona	
Problem	
Task / Constraint	
How is it addressed now? (Pre- TRUSTS)	
Other Personas	
Problem	
Task / Constraint	
How is it addressed now? (Pre- TRUSTS)	

Describe the Expected Benefit

*Describe the benefit that each persona hopes to achieve from the UC (**after** TRUSTS is implemented). Please try to be specific on the benefits that may apply … Cost? Time? Agility? Safety? Security?*

End user personas	
Describe benefit	
Specific benefit	*Quantify the potential benefit*
Cost reduction?	
Revenue Increase?	
Time saved?	
Faster Time-to-Market?	
Safety?	
Security?	
Accessibility?	
Persona experience?	
Other …	

App. Provider Personas	
Describe benefit	
Specific benefit	Quantify the potential benefit
Cost reduction?	
Revenue Increase?	
Time saved?	
Faster Time-to-Market?	
Safety?	
Security?	
Accessibility?	
Persona experience?	
Other …	
Other Provider Personas	
Describe benefit	
Specific benefit	Quantify the potential benefit
Cost reduction?	
Revenue Increase?	
Time saved?	
Faster Time-to-Market?	
Safety?	
Security?	
Accessibility?	
Persona experience?	
Other …	

Marketplace Expectations
Please attempt to define what are the expected (required or nice to have) functionalities provided by the TRUST data Marketplace which will benefit in a business level the involved parties of the UC.

Functionality	Beneficiary	Required	Nice to Have

Moreover, the template for collecting the functional requirements from the relevant partners and end-users of the data marketplace along with the UC needs, has been defined. The following Table presents the template for listing the functional requirements collected and given in D2.2.

Table 4 presents the template for listing those functional requirements, providing the main description of the actual requirement elicited, a unique identifier number for each requirement, the pertinent current marketplace, questionnaire, interviews and use cases requirement references as well as the involved tasks for its implementation. This Table is completed and described in detail in D2.2 "Industry specific requirements analysis, definition of the vertical E2E data marketplace functionality and use

cases definition I". For a reference we are also presenting a sample on how the Table shall be completed.

Table 3: Functional Requirements Template

Req. ID	Description	Requirement reference	Tasks	Applicable / Needed	Non-Applicable/ Not needed
Service Description					
FRXX	Functional requirement description	UCXX, etc.	TX.X		
FRXX	Functional requirement description	UCXX, etc.	TX.X		

Following the business validation on UCs and their functional requirements, a necessary input for the creation of the Business models will be provided (as it is described in DoA) in T7.1 via the Business Validation Template. So, among the business validation of the UCs needs, what is planned to be performed here, is the iterations during trials where the UCs will be used among other to validate TRUSTS Data Marketplace. The business model will be developed by applying tools for business model innovation as developed in TUD's award-winning platform businessmakeover.eu[9]. The tools will be applied in workshops with project participants and, later on in the project, outside stakeholders. To inform the business model development, first, through desk research and interviews, a range of potential data marketplace business models will be explored, leading to a taxonomy of possible business model design options. In doing so, this task will closely interact and leverage outputs of "T2.1 EU and worldwide data markets".

Furthermore, along with the collaboration with WP3 (Platform Implementation), there is also a collaboration with WP4 with respect to the business validation to be performed on its respective task. To this end a questionnaire has been unified as presented in Table 4. The purpose of this questionnaire is to align and clarify the purpose of WP4 with the UC's definitions and requirements.

The answers / feedback (provided by the UC owners) in this questionnaire will affect the work needs to be performed in WP4 related to the following topics and Tasks, and additionally will affect the platform architecture related to T2.4 as well as the platform development and integration related to T3.5

- Privacy Preserving Data Analytics (T4.1);
- Privacy Preserving Transfer Learning and Classification (T4.2);
- Anonymisation and de-anonymisation (T4.3);
- Federated Deep Learning methodologies (T4.4);
- Transformation of algorithms to privacy-preserving certified (T4.5).

---

[9] https://businessmakeover.eu/

All questions listed in Table 8, have been identified in collaboration with the WP4 and will further enhanced with additional questions, if this is required during the project lifecycle.

Table 4: Business Questionnaire Template – WP4 Alignment

WP4 Questions & Business Validation	UC1 Answer	UC2 Answer	UC3 Answer
What is the approximate amount of data expected to have in each data set?			
Is the data public or private?			
What kind of data are we going to have in the TRUSTS marketplace? (Full data, metadata etc.)			
Will access be given to all the use-case datasets?			
For each use case - How often/frequently do the datasets are being updated?			
Will data be privacy preserved by means of the platform or at the bank, insurance company, etc.?			
Are the database attributes going to be fixed and standardized, or will the data providers (sellers) decide on which attributes to input to the market?			
Many member states of the European Union do not use the Euro as currency. So, is the market designed to provide insights on financial data with different currencies? Or is it a further step for after the implementation is done?			
How is the communication between data sellers and data customers (buyers) going to take place in the market? Will it be assisted in order to avoid confidential information leaks during the interactions? For instance, in UC 3, there will be a chatbot to allow the communication, but how about UC 1 and UC 2?			
How do you see the objective of WP4 "this WP is to investigate, design and improve cryptographically secure protocols that enable data analysis of privacy-sensitive data." integrated with your use case? (How does your use case related to data analysis of privacy-sensitive data ?)			

4.2   TRUSTS Business Validation and KPIs

A number of Key Performance Indicators (KPIs) have been identified and listed in the DoA. This detailed set of KPI metric parameters will be considered so as to give validation results from the business point of view of each UC as listed in Tables 5-7. Threshold limits (as "Target Value in M36") of the results are also defined per target KPI, along with an initial refinement from the UC owners. During the 1st Business Validation period, calculation and validation method will also be defined, as well as refinements and any additional KPIs if those are needed.

Those validations are important since will be the corresponding outputs that will allow to identify the UCs that have the highest commercialisation potential in order to progress to the next step of creating a data marketplace service portfolio.

In addition to the business validation KPIs which are currently based on UC (as those are listed in DoA), and since we have selected an agile evaluation methodology, an intermediate set of KPIs might also be defined (by UC participants and related partners) by M18 as well, if the Consortium foreseen to. This should be in align with MS3 (First Pilot Deployment) and the additional KPIs might be measured during the 1st trials. In case this applied, the below Tables 5-7 will be amendment accordingly.

In addition, we believe that the business validation should not only be based on the UC but to the perception of the external world as well, that's why we propose to have a feedback loop from the advisory board and dissemination activities.

Table 5: Key performance indicators (KPIs) for Use Case 1

KPI	Baseline Value	Target Value (M36)	Calculation Method	Validation Method	Validation Results
Number of alerts per scenario	Number of alerts per scenario issued by WiseBOS ERP solution	Decreased by 50% from baseline			
Detection accuracy	Detection accuracy from WiseBOS ERP solution	Increased by 50% from baseline			
Number of false positives	Number of false positives flagged by WiseBOS ERP solution	Reduced by 30% from baseline			
Number of false negatives	Number of false negatives flagged by WiseBOS ERP solution	Reduced by 30% from baseline			
SAR[10] capture	70%	>95%			
Losses due to	As per self-assessment from end-	Reduced by 30%			

---

[10] A Suspicious Activity Report (SAR) is a document that financial institutions must file with the Financial Crimes Enforcement Network (FinCEN) following a suspected incident of money laundering or fraud.

fraud	users	from baseline			
Number of data providers interacting with the Platform	2 at the start of the use case	Minimum 10 by M36 (+400%)			
Number of end-users interacting with the Platform	1 at the start of the use case	Minimum 10 by M36 (+400%)			

Table 6: Key performance indicators (KPIs) for Use Case 2

KPI	Baseline Value	Target Value (M36)	Calculation Method	Validation Method	Validation Results
Number of target marketing analysis	2 per month	>10 per month			
Data readiness for correlation	Low (1 week for data to become ready)	High (1 day for data to become ready)			
Data valuations	2 per month	>10 per month			
Data anonymizations / deanonymizations	<1 per month	>10 per month			
Number of data providers interacting with the Platform	2	>10			
Number of end-users interacting with the Platform	2	>10			

Table 7: Key performance indicators (KPIs) for Use Case 3

KPI	Baseline Value	Target Value (M36)	Calculation Method	Validation Method	Validation Results
Decrease (X%) operational cost for	Decrease (estimated at	Decrease (estimated at 20-			

the same collectability

Base line will be taken during analysis phase from the Creditor, to register current KPI metrics (AS IS) and to be able to compare with new results (TO BE).

Final measurements of KPI needs the solution to be installed at production and run for a period in order to fine-tune and afterwards measure the KPIs | 5%) operational cost for the same collectability 5% | 25%) operational cost for the same collectability | | | |
| Increase (X%) efficiency and productivity

Base line will be taken during analysis phase from the Creditor, to register current KPI metrics (AS IS) and to be able to compare with new results (TO BE).

Final measurements of KPI needs the solution to be installed at production and run for a period in order to fine-tune and afterwards measure the KPIs | The human agent's efficiency will be increased by 5% with the help of the Virtual Assistant. | The human agent's efficiency will be increased 15% with the help of the Virtual Assistant | | | |
| Cost reduction (X%) for process costs on debt management services | Decrease in wealth management operational | Decrease in wealth management operational costs | | | |

Base line will be taken during analysis phase from the Creditor, to register current KPI metrics (AS IS) and to be able to compare with new results (TO BE). Final measurements of KPI needs the solution to be installed at production and run for a period in order to fine-tune and afterwards measure the KPIs. *needs to be reviewed	costs (through a 20% increase in process automation).	(through a 40% increase in process automation).			
Complaints Rate KPI Base line will be taken during analysis phase from the Creditor, to register current KPI metrics (AS IS) and to be able to compare with new results (TO BE). Final measurements of KPI needs the solution to be installed at production and run for a period in order to fine-tune and afterwards measure the KPIs. *needs to be reviewed	Decrease of 5% to 10%	Decrease of 5% to 10%			
Process automation increased (X%) Base line will be taken during analysis phase from the	Estimated increase in efficiency and productivity by over 15%	Estimated increase in efficiency and productivity by over 25%			

Creditor, to register current KPI metrics (AS IS) and to be able to compare with new results (TO BE). Final measurements of KPI needs the solution to be installed at production and run for a period in order to fine-tune and afterwards measure the KPIs					
Increase (X%) collectability of debt Base line will be taken during analysis phase from the Creditor, to register current KPI metrics (AS IS) and to be able to compare with new results (TO BE). Final measurements of KPI needs the solution to be installed at production and run for a period in order to fine-tune and afterwards measure the KPIs.	Estimated increase in collectability of debt by 10%	Estimated increase in collectability of debt by 20%			
Improve (X%) at default predictability Base line will be taken during analysis phase from the Creditor, to register current KPI metrics (AS IS) and to be able to compare with new results (TO BE).	Foreseeing the end-customer's probability to default in at least 20% of the cases.	Foreseeing the end-customer's probability to default in at least 60% of the cases			

Final measurements of KPI needs the solution to be installed at production and run for a period in order to fine-tune and afterwards measure the KPIs.					
Number of data providers interacting with the	1 at the start of the use case	Minimum 3 by M36			
Number of end-users interacting with the Platform	1 at the start of the use case	Acquisition 3 customers by M36			

5   Technological Validation

This section describes the methodology for the technological validation in TRUSTS. The objective is to validate the three UC's technical wise and develop user acceptance test's templates for the UC's so as to provide the potential highest commercialization.

As mentioned in previous chapters, section 3, we have presented a process where technological and business validation goes hand in hand, starting with templates for the business validation before the technological validation. Thus, we focus on methods for confirming that there is real business value, and how this can be reflected in business metrics.

In this section an analysis will be given with respect to the Test-Driven Development Methodology (TDD) as part of the technological validation of the TRUSTS data marketplace along with a reference to the unit and acceptance tests and templates to be used during the life cycle of the project.

Technological validation endorses that integral technologies can be absorbed into a complete system solution and its performance and operation are met under predictable operating scenarios.
Before transfer to the market the technology must be validated empirically by simulating its future practical use. Technology prototypes at a first level are explored in simplified situations, and these simulations are extended to conditions of practice step by step as more becomes known about the technology. When scaling up to practice, analysts want to derive from validation to practice.

In TRUSTS technological validation refers to the technical and interoperability testing issues related to solutions/applications developed in the framework of the data marketplace based on TDD Methodology and the indicative user acceptance tests and templates as per chapter 5.2.

5.1   Test-Driven Development (TDD) Methodology

TDD is an evolutionary software approach to development process that relies on the repetition of a very short development cycle and where the requirements are turned into very specific test cases, then the code is improved so that the tests pass. TDD starts with designing and developing tests for every small functionality of an application. In TDD, the test is developed first and specifies and validates what the code will do. The primary goal of TDD is to make the code clear, simple and bug-free.

In other words, the simple concept of TDD includes test-first development, which means that the developer first writes a fully automated test case before writing the production code to fulfill that test and refactoring. Hence, this is helpful by avoiding duplication of code as we write a small amount of code at a time to pass tests before the actual development of the application. As a matter of fact, the more you repeat prior to disposition, the more successful are the benefits of TDD, as unit tests are applied often, in order to make sure that new functionalities do not deprave old ones.

TDD methodology will be followed during the implementation lifecycle of the project and consist of the following steps leading to the technological validation. The process of the TDD Methodology is exposed in Figure 17.

- **Write / re-write tests:** Write a single unit test that succinctly defines a function or improvements of a function. Functions should be kept simple and should only solve a single separated task and not mixing to much functionality into one function as per KISS principle[11]. Understand the feature's specification and requirements to write the test in a testing

---

[11] https://marketbusinessnews.com/financial-glossary/kiss-principle/

framework suitable to the software environment. This could be a modified version of an existing test. The TDD approach makes the developer focus on the requirements before writing the code unlike writing unit tests after the code is written.

- **Run all tests and see if the new test fails:** TDD validates that the test harness is working correctly and shows that the new test does not pass without requiring new code because the required function already exists. It eliminates the possibility that the new test is flawed and will always pass. The new test should fail for the expected reason.
- **Write production code:** Write code that is just enough to cause the test to pass. Keep the code as simple as possible. It does not have to be the most perfect piece of code as long as it can solve the initially failing test. Do not write code beyond the functionality that the test checks or requires.
- **Run tests:** Run and re-run all tests until all cases pass to ensure that the newly added code meets the test requirement without breaking functionality or degrading any features. If test cases do not pass, the new code must be optimized until they do. Write tests with minimum code changes after every test runs to ensure that good unit test coverage for the software will eventually add up to the overall quality of the product.
- **Clean up / Refactor code:** Refactoring is a crucial step between each pass and every next failure. Refactoring is essential to make sure that the code conforms to the simplicity criteria while it maintains functionality. Move the new code from where it was convenient for passing a test to where it more logically belongs to keep your code neat and agile. Remove duplication and aim to improve the software's readability and maintainability. Continually re-run the test cases throughout each refactoring phase to ensure that the process will not alter any existing functionality or to confirm that no additional bugs were introduced.
- **Repeat:** Repeat the cycle for continuous integration pushing functionality forward. Have as few as 1 to 10 edits between each test run and keep the size of the steps small. This is to easy to undo or revert the code if it does not instantly satisfy the new test, or other tests fail unexpectedly. It will help reduce the need for debugging in the later parts of the development cycle.

Figure 7: Test Driven Development Analysis12

In regards to the benefits of TDD, it gives a way to think via one's requirements or design before the developer writes functional code. It is also a programming technique that enables the developer to take

---

[12] https://frameworkltc.com/blog/tag/Test+Driven+Development

a small step while building software and it is more productive as compared attempting to code in huge steps. Thus, TDD makes the code simpler and clearer, allowing the developer to sustain less documentation Furthermore, TDD endorses affirmative testing of your application code and detailed operational requirement, plus, both acceptance tests (detailed specifications) and unit tests are inputs for TDD.

In TRUSTS, the test process will fit neatly into the project's iterative agile development process in WP3 and allow for implementation of a Test-Driven Development (TDD) methodology incorporating unit tests and acceptance tests while tests will be performed as part of the platform and UCs evaluation in WP5. The methodology will also define how the interaction with the vertical end-users will be achieved taking into consideration the specifics of T2.1.

5.2   Unit Test, User Acceptance Test and Templates

Unit test are an approach of software testing where individual units/components of software are tested. The goal is to validate that each unit of the software acts as designed. A unit is the smallest testable part of any software. It normally has one or a few inputs and typically a single output. In procedural programming, a unit may be an individual program, operation, method, etc. In object-oriented programming, the slightest unit is a method, which may belong to a base/ super class, conceptual class or derived/ child class. Unit testing frameworks, along with drivers, stubs, and mock/ fake objects are usually used to help in unit testing

Among several benefits' unit tests are valuable for the following reasons:

- Raises confidence in alternating/ sustaining code,
- The cost of amending a defect exposed during unit testing is smaller in comparison to that of defects discovered at higher levels,
- Finds software bugs at an early stage,
- Helps to simplify the debugging process,
- Reduces the cost of bug fixes,
- Code is more trustworthy through code is improvements,
- Facilitates changes and simplifies integration by allowing the programmer to proceed with the refactoring of the code or upgrade system libraries at a later stage to make sure that modules are still working correctly,
- Provides documentation of the system,

With respect to the user acceptance tests, similar to a unit test an acceptance test usually has a dual result, either to pass or to fail. Normally the detection of a failure does not prove the presence of a defect in the product.

 Thus, the user acceptance tests could be characterized as a formal description of the behavior of a software product, normally expressed as an example/usage scenario. A number of different systems and approaches have been proposed for such examples. In several cases the goal is that it should be feasible to automate the execution of such tests by a software tool, either ad-hoc to the development team or off the shelf. An acceptance test is normally expressed as an example/usage scenario. A number of different systems and approaches have been proposed for such examples. In several cases the goal is that it should be feasible to automate the execution of such tests by a software tool, either ad-hoc to the development team or off the shelf.

Acceptance testing has the below benefits, accompanied by those which can be gained from unit tests:

- Encourage closer collaboration/ communication between the three of the vendors, the developers from one side and the customers/ users or domain experts from the other side, as they involve the business requirements that should be expressed
- The quality criteria of the product are defined in the early phase of development/implementation
- Provide a clear "contract" between customers and developers, given the fact that a product passing acceptance tests will be considered adequate.
- Decrease the chance and severity both of new defects and regressions
- The engineering team ends up minimizing the pressure during the implementation and risks of post-implementation live fixes
- Clients satisfaction is increased, as they are more confident that the requirements are met
- Stakeholders use the information gathered through User Acceptance Testing to better understand the needs of the target audience.

The work flow of unit and acceptance tests is illustrated on the Figure 9 below:
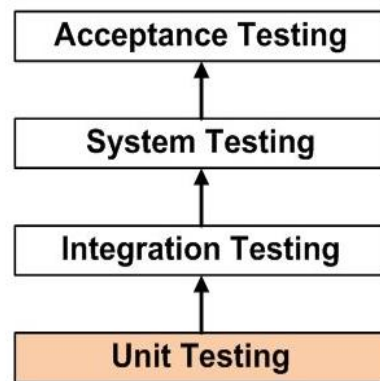
Figure 8: Acceptance Test and Unit Test workflow

Specific technical validation and interoperability testing plans will be developed prior of each testing phase and for each prototype, in close collaboration with all the relevant development teams, so as to detail the goals, the indicators and the approaches to be followed. The UC's are also taken into account, as they indicate the targeted user experience when using the platform and the offered services.

Testing are carried out by the development teams of each UC and should be conducted before the iterations of the user testing of TRUSTS in each phase. This will allow adequate time for debugging, if needed, before the end user's experience with the applications. In case of significant technical failures, then the respective prototypes will move to the next user test iteration. The findings of this activity will be continuously fed back to the respective development teams. In this way, from each evaluation round the results from the users and also from the developers' point of view will be accumulated in order to lead to further optimization in the next version of the prototype or application. In each technical validation iteration, the versions of the prototypes will be assessed in terms of technical performance against the initial functional requirements as per D2.2.

The aspects to be estimated may include:

- **Functionality Testing**: Assessment for its correct functioning according to its functional and technical requirements.
- **User Interface Testing**: Evaluation in respect to its operation, content navigation, etc.

- **Interaction Testing**: Assessment for errors that may interact with other modules developed in TRUSTS
- **Compatibility Testing**: Evaluation for compatibility with different devices (e.g. smartphones, PC's), diverse OSs (e.g. Windows and Linux) and various browsers (e.g. IE, and Firefox)
- **Performance Testing**: Assessment for its performance for diverse Internet connection speeds, how its response to different devices, OSs and browsers and stress testing
- **Security Testing**: Estimation for unauthorized access to information, unsecured provision of private data etc.

Furthermore, as part of technological validation, a testing architecture will follow in D2.6.

5.2.1 TRUSTS Technological Validation via UATs

In TRUSTS, user acceptance test (UAT) template will be incorporated for the implementation of a TDD methodology to fit neatly into the project's iterative agile development process. The methodology will entail acceptance test procedures for conducting both the technological and business validation of the UC's considering the associated service management. The UAT template helps testers to identify, define and execute UAT test cases based on defined requirements. It ensures also that all relevant information is available for the person conducting the UAT test cases.

Furthermore, in TRUSTS the Unit Acceptance Tests (UAT's) will be used by the UC participants in order to check if the client needs are met with the developed solution. Table 7 presents the defined UAT template used during the life cycle of the project.

Table 8: Technological Validation – UAT Template

UAT Scope (In Scope – Out of Scope)			
UAT - In Scope		**UAT - Out of Scope**	
In Scope *List features that are tested.*		Out of Scope *List features that are not tested.*	
UAT Assumptions and Constraints			
UAT Assumptions			
Assumption *List the UAT assumptions.*			
UAT Constraints			
Constraint *List the UAT constraints.*			
UAT Risks			
Description	**Probability** **High | Medium | Low**	**Impact** **High | Medium | Low**	**Mitigation**
Risk *List the risks of UAT.*	*How likely is the risk to occur?*	*What is the impact of the risk on the UAT?*	*Steps to avoid the risk.*
UAT Team Roles & Responsibilities			
Name	**Roles**		**Responsibilities**
Name *List names of people involved in testing.*			
Name *Add more rows if needed.*			
UAT Entry Criteria			
Criteria			
Entry Criteria *Factors that must be present to enable the start of the UAT.* **Example**: *Testing environment/ data is available.*			
UAT Requirements-Based Test Cases			

Test Cases			
Test Case *Identify the test cases along with the expected results.* *Example*: *Test Procedure:* *Login with a corporate user account.* *Username: abc / Password: abc* *Expected Results:* *An error will be displayed for the wrong credentials.*			
UAT Test Results			
Test Cases	**Pass/Fail**	**Tested By**	**Date Tested**
Test Case *Name the test case.* *Example*: *Test Procedure:* *Login with a corporate user account.* *Username: abc / Password: abc* *Expected Results:* *An error will be displayed for the wrong credentials.*			
Addendums & Appendices			
Include any additional documents.			

Analyzing a bit more the above UAT Template, we have the following information which are required:

- **UAT Scope.** First of all, we need to define the UAT scope of our particular test case by defining a list of features that we want to test, and the ones that are not being tested.
- **UAT Assumptions and Constraints.** After the definition of our UAT scope, we need to be aware of the assumptions and constraints of the test. These assumptions and constraints include timing and available resources, as well as test documentation processes. Some constraints could be the operating system to be used and specific browser versions. In addition, we must also clearly state the defined assumptions, such as how our test environment looks like, or how a tester should handle the error reporting.
- **UAT Risks.** As UAT is a very crucial part of the software development cycle, we must look at potential UAT risks while planning, executing and analyzing our UAT test case. In our example we could face some of the following risks during UAT:
  - o UAT testers might not be properly trained and don't have complete knowledge of the business and user needs.
  - o Due to lack of resources and time constraints, tests might be incomplete by the defined deadline.
  - o The testers might not know how to properly report errors and bugs while testing.
  - o Sometimes the developed features are incomplete and lack too many bugs which do not allow a test case to be completed at all.
- **UAT Team Roles & Responsibilities.** Participants of a UAT team can vary from UC to UC. The entire UAT team will be responsible for coordinating the preparation of all test cases and its execution. The UAT team will ensure that defined test cases are planned and conducted accordingly, and that test results are documented and shared among the project team.
- **UAT Requirements-based Test Cases.** While all previous steps aim to ensure a well-executable test case, it's now time to describe the actual test case itself to be successfully conducted by the UAT testers.
- **UAT Test Results.** In order to execute a test case, testers follow the described test cases step by step. Ideally, the tester can successfully execute the test. However, one important part of UAT testing is to report unexpected system behaviors, or even real bugs. After successfully

completing the UAT test cases, every tester must provide their test results. A well-documented UAT test case enables the product or project team to conclude on the next steps and define next actions depending on the test results.

6   QoE, QoS and SUS Validation Methodology

In the section below an emphasis is given to the Quality of Experience (QoE) and to the Quality of Service (QoS) based on the SUS scoring methodology, as part of the entire validation process that aims to be followed during projects' lifecycle for the Platform and the offered services in the framework of commercial value testing and user's experience.

The major requirement here is for the project to deliver outputs that have commercial value and potential based on the QoE so as to measure objective data marketplace user experience and the QoS that will give a qualitative measurement of test execution.

6.1   Quality of Experience

In recent years, Quality of Experience (QoE) has become more and more essential to service providers. A poor QoE can lead to the loss of reputation, decreased customer loyalty and ultimately missed revenues[13].

The user's QoE has become valuable of performance measures as it directly relates to how a user judges the service provided. The worth to the service provider in rating the QoE metric is that the quality of service is not a system generating technical measures but a user's subjective opinion on how they experienced the quality of the service on offer.

The QoE measures total system performance using subjective and objective measures of customer satisfaction. It is a concept, similar to the field of user experience, but with its roots in telecommunication. QoE is an emerging multidisciplinary field based on social psychology, cognitive science, economics, and engineering science, focused on understanding overall human quality requirements[14].

Hence QoE differs from QoS, which assesses the performance of hardware and software services delivered by a vendor under the terms of a contract. It could be also mentioned that QoE depends on customer experience, assessments are compiled from large user group polls. Some of the success factors of ratings of QoE are: efficiency, reliability, security, usability, cost etc.

In regards to TRUSTS a significant role will be given to QoE so as to evaluate the data market place along with the use cases in order to deliver outputs that have commercial value and potential.

QoE will provide a gauge objective data marketplace user experience to validate the TRUSTS data marketplace throughout the concept of SUS Methodology evaluation scale[15] that is presented at the end of this section.

6.2   Quality of Service

Quality of Service, (QoS) is one of the most important competitive factors in today's business landscape. Quality of Service (QoS) controls and handles network resources by putting priorities regarding specific types of data on the network. It can be also defined as the measurement of the whole performance of a service such as telephony or computer network or a cloud computing service, most concrete the ones seen by the users of the network. Thus, QoS can be characterized as a set of techniques to manage network resources and as a capacity to provide different priority to different applications, users, or data

---

[13] https://bics.com/quality-of-experience-why-does-it-matter-to-capacity-service-providers-and-what-affects-it/
[14] https://en.wikipedia.org/wiki/Quality_of_experience
[15] https://www.usabilitest.com/system-usability-scale

flows, or to guarantee a certain level of performance to a data flow. This enables the network administrator to appoint the order to handle packets, and the amount of bandwidth sustained to that application or traffic flow. In order to quantitatively measure quality of service there are several parameters such as, usability, availability, security, etc.

Whatever can be defined as a service, it has quality if it can fulfil the customer's needs and demands, and the adequate service to be consistent with customer expectations or above them.

 There are several common factors of service quality such as:

- **Process Quality**: Refers to production quality approaches and the processes to provide services to customers.
- **Product Quality**: calculated after the provision of the adequate service. In fact, product is what the customer gets from marketplace.
- **Interactive Quality**: Relegates to interactions between the customers and the service providers.

Additionally, there are various reasons in regards to the necessity of paying attention to service quality in order to provide high quality services, for example:

- **Increasing Customer Expectations**: Related to increase customers' awareness and knowledge, and competitors' performance.
- **Environmental Factors**: In nowadays due to the expanded access to the internet, customers can easily achieve to extract information from around the world.
- **Services Nature**: Acknowledge of attributes of services for end users/recipients.
- **Benefits Arising from Service Quality**: Increasing the organization's ability to provide efficient services for customers because organization identified the demands and needs of their customers. Thus, data market profit will increase by building up the efficiency and effectiveness in providing services.

6.3 SUS Validation Methodology

The System Usability Scale (SUS), is an additional methodology utilized in order to enable the overall scoring on a number of validations metrics related to the TRUSTS Data marketplace and the offered services. The SUS is a widely used self-administered tool for the evaluation of usability of a wide range of products and user interfaces. The principal value of the SUS is that it provides a single reference score for participants' view of the usability of a product or a service[16].

Furthermore, usability assessment is an important part of the overall design and development of a product or service, which consists of iterative cycles of prototyping, design and validation. Ideally, the usability evaluation must be present at all stages of the design and development process, and must be iterative in order to allow a continuous improvement of the results[17].

SUS is a standardized questionnaire designed to assess perceived usability. The standard version of the SUS has 10 items, each with five steps anchored with "Strongly Disagree" and "Strongly Agree."

---

[16] https://www.sciencedirec.com/science/article/pii/S1877050915031191

[17] European Portuguese Validation of the System Usability Scale (SUS)
https://www.sciencedirect.com/science/article/pii/S1877050915031191

6.4 TRUSTS Validation via QoE, QoS and SUS

With respect to the validation of TRUSTS data marketplace a significant emphasis will be given to QoS and QoE as a complementary process for the overall evaluation of the platform and the offered services.

For this purpose, a set of metrics have been selected based on the QoE and the QoS.

Regarding the QoE of TRUSTS Platform the following metrics have been defined:
- **Usability**: determines the quality of the use of a product or service, specifically the efficacy, efficiency and easiness of use of a certain interface/system.
- **UI Design**: presents the overall appearance, structure and designation of the platform.
- **Operations Completeness**: refers to the successful onboarding process of the offered services and data/metadata as well as to the federation issues that should be tested.
- **Service Excellence**: presents the act of surpass customer expectations and delivering an industry-leading experience which really fulfills the clients' needs and helps the end users.

With respect to the QoS, as part of the commercial value and user's experience framework, an essential role has been given to the evaluation of the data marketplace as well as the different use cases related to the services of the marketplace so as to deliver outputs that have commercial value and potential considering all the above.

TRUSTS data marketplace envisages to provide different kind of services that will have quality and satisfy end users' needs and demands. Accepted factors of service quality as process quality, product quality etc. have a significant role in order to create the TRUSTS marketplace portfolio.

QoS should give a qualitative measurement of test execution throughout the following metrics:

- **Usability**: refers to the quality of use as a level of valuation presenting if a product satisfies the stated and implied needs of the users in the data marketplace.
- **Contract Quality**: provides clear information about data usage and information on data delivery, acceptance and payment.
- **Security:** refers to the use of effective security methods and mechanisms for service and data security under GDPR compliance.
- **Easy setup**: refers on how easily, fast and accurate a service or data provider can proceed with the onboarding of its service or data to the data marketplace.
- **Automated Confirmation Transaction (ACT):** designed to document and report the clearing of trades in the data marketplace providing faster access to trade information, increasing the efficiency of trade reconciliation and back-office transactions and provides online access to the transaction logs.
- **Help Support:** refers to a number of ways you can receive support from the provider, i.e.: a help center, live chat 24/7.
- **Search Engine Optimization (SEO):** refers to a hidden engine of marketplace growth as it consists of various activities and techniques that help your platform content to appear as high as possible in search engine results, when people search topics using specific keywords. It enhances the platform's discoverability, readability, and usability.
- **Service Update Notification:** provided an updated information for each new service is available on the data marketplace to notify existing users. This could be done either via a pop up once or via an email notification.

Following QoE and QoS in TRUST, partners will validate the data marketplace and the offered services, as a separate process (via the UC trials) along with the processes and steps needed for each one of them. By doing that, the end-users validate the marketplace as viable and the offered services as adequate.

For achieving this scope, an adequate questionnaire in the form of a survey will be provided to all partners and especially to the UC's owners in order to measure the evaluation of each process and step of the implementation of the TRUSTS platform and its portfolio. This evaluation survey will be held as presented in Figure 10-11 taking into consideration all the predefined processes i.e.: from service on boarding to service usage including all the relevant steps of the usage of the platform at each phase as those are shown on the Table 9.

The following initial indicative steps (which were identified under WP2), will be used in the first processes phase for the uploading of the services in the platform as well as for its usage by the end-users. During the life cycle of the project and as part of the TRUSTS validation, those processes will be more concretely defined and amended accordingly.

Table 9: Business Processes to be Validated

Process	Steps
Service on-boarding	Service files (code) uploading
	Service test data uploading
	Smart contract formatting
	Terms & conditions formatting
	Service on-boarding completion and testing
	Service added in marketplace catalogue
Data on-boarding	Service selection
	Datasets uploading
	Smart contract formatting
	Terms & conditions formatting
	Datasets on-boarding completion and testing
Service inclusion in the marketplace portfolio	Link of relevant data / metadata to the service
	Launched in marketplace
	New service availability announcement
User enrolment	Company enrolment process
	User verification and approval (by Admin)
	User (per company) enrolment process (multiple profiles)
	User type selection (service provider / buyer)
	Acceptance of terms & conditions
	User verification and approval (by admin)
	User access in the marketplace
Service usage	User login
	Search in service catalogue
	Find the requested service
	Subscription selection
	Sign the contract
	Acceptance of terms & conditions
	Service end2 end usage
	Compensation billing
	Transaction logs / SLA management
TBD other processes to be validated	TBD other process steps to be validated

TRUSTS will have a consistent set of operational functionalities so as to be able to provide the ability of offering sustainable and operational business services and datasets, carrying out the following operational performance under Privacy and GDPR processes:

- **Service on-boarding process:** Services will be installed in the TRUSTS data marketplace of offered through a respective API. The on-boarding process includes the following: testing of technical, performance and security aspects, smart contract establishment, services catalogues, the search engine.
- **Data/Meta data on-boarding process:** The TRUSTS data marketplace shall allow the on-boarding of data, by choosing the appropriate service to be used under a specific file type supported by the service. Onboarding data includes the following: testing of data, performance and security aspects, smart contracts.
- **Service inclusion in the marketplace portfolio:** service for updated notifications with respect to the offered service. Both services and data provided in TRUSTS data marketplace, after on-boarding, shall be included in the marketplace catalogue and search engine, after following an approval process. Hence the user can search for them either in direct way or by using a number of keywords in the platform's search tool.
- **User Enrolment:** Subscription and administration process and contracting with client companies/ users (per company) for enrollment process following acceptance of terms and condition, user verification and approval of access to the marketplace by the administrator.
- **Service usage:** Billing, reimbursement of affected parties in the value chain, revenue assurance, transaction logs.

With respect to the results those will be consolidated and presented after the execution of the indicative questionnaire in a table showing the scoring of each process and relevant steps of the marketplace evaluation.

The adequate toolset for validating the TRUSTS data marketplace as well as the offered services took into account the respective steps, methods and parameters along with the business and technological metrics divided into QoS and QoE for the purposes of the overall Platform evaluation and the nominated services.

In order to measure the evaluation of each process and step of the marketplace and in parallel with the survey as per Figure 10-11, we took into consideration all metrics regarding QoS and QoE, utilizing the existing and predetermined questions as Table 10 with respect to the SUS Methodology for complementing the qualitative evolution of the Platform and its portfolio. As previously mentioned, these validation methods will be used by the partners, especially UC participants during the different phases of the TRUSTS Platform implementation. Thus, the SUS Standardized Questionnaire along with the marketplace survey will be used in parallel with the defined business and technological validation templates during the life cycle of the project.

Table 10 presents the standardized SUS questionnaire which will be used mainly by the UC participants, in combination with custom questionnaires and/or surveys for complementing the qualitative evolution of the data marketplace and its portfolio. The goal is to validate that the implemented solution is viable and TRUSTS services are sufficient.

Table 10: SUS Standardized Questionnaire

		The System Usability Scale Standard Version		Strongly Disagree				Strongly Agree
				1	2	3	4	5
	1	I think that I would like to use this system frequently.		O	O	O	O	O
	2	I found the system unnecessarily complex.		O	O	O	O	O
	3	I thought the system was easy to use.		O	O	O	O	O
	4	I think that I would need the support of a technical person to be able to use this system.		O	O	O	O	O
	5	I found the various functions in this system were well integrated.		O	O	O	O	O
	6	I thought there was too much inconsistency in this system.		O	O	O	O	O
	7	I would imagine that most people would learn to use this system very quickly.		O	O	O	O	O
	8	I found the system very awkward to use.		O	O	O	O	O
	9	I felt very confident using the system.		O	O	O	O	O
	10	I needed to learn a lot of things before I could get going with this system.		O	O	O	O	O

Additionally, Figures 11-14 presents screenshot of the marketplace validation questionnaire, (designed using SurveyMonkey[18] cloud-based tool) with respect to the first process which is the 'Service on-boarding' in reference to QoS and QoE metrics. The same applies for the remaining four processes and the steps corresponding to each one of them and it will be adjusted if any amendment will take place regarding the processes during the life cycle of the project.

The scoring methodology, follows the architecture thinking of SUS approach with the necessary adjustments in an evaluation method from 1 to 5, where 1 means "strongly disagree" and 5 means "strongly agree". An odd scale is preferred so to be in align with SUS methodology.

---

[18] https://www.surveymonkey.com/

Figure 9: QoE Marketplace Validation Questionnaire

Figure 10: QoS Marketplace Validation Questionnaire

The entire set of questionnaires will include all the information presented in Tables 11 -12 (only cells colored as green), in relation to QoE and QoS. Those indicative parameters have been chosen through research and by laying out a standardized evaluation approach in order to establish this based on various parameters. During the life cycle of the project and as part of the commercial value testing and users experience, those parameters will be updated and finalized accordingly.

Table 11: Marketplace Validation Based on QoS Evaluation Parameters

Marketplace Evaluation										
		Quality of Service								
Process	Steps	Usability SUS (Score 10-100)	Contract Quality (Score 10-100)	Security (Score 10-100)	Easy set up (Score 10-100)	Automated Confirmation Transaction (ACT) (Score 10-100)	Help Support (Score 10-100)	Search Engine Optimization (SEO) (Score 10-100)	Service Update Notification (Score 10-100)
Service on-boarding	Service files (code) uploading		N/A			N/A	N/A	N/A	N/A
	Service test data uploading		N/A			N/A	N/A	N/A	N/A
	Smart Contract formatting					N/A	N/A	N/A	N/A
	Terms & Conditions formatting					N/A	N/A	N/A	N/A
	Service on-boarding completion and testing		N/A			N/A	N/A	N/A	N/A
	Service added in marketplace catalogue		N/A			N/A	N/A	N/A	N/A
Data on-boarding	Service selection		N/A			N/A	N/A	N/A	N/A
	Datasets uploading		N/A			N/A	N/A	N/A	N/A
	Smart Contract formatting					N/A	N/A	N/A	N/A
	Terms & Conditions formatting					N/A	N/A	N/A	N/A
	Datasets on-boarding completion and testing		N/A			N/A	N/A	N/A	N/A

Service inclusion in the marketplace portfolio	Link of relevant data / metadata to the service		N/A		N/A	N/A	N/A	N/A	N/A
	Launched in marketplace		N/A		N/A	N/A	N/A	N/A	N/A
	New service availability announcement		N/A		N/A	N/A	N/A	N/A	
User enrolment	Company enrolment process		N/A		N/A	N/A			N/A
	User verification and approval (by Admin)				N/A	N/A			N/A
	User (per company) enrolment process (multiple profiles		N/A		N/A	N/A			N/A
	User type selection (service provider / buyer)		N/A		N/A	N/A			N/A
	Acceptance of Terms & Conditions				N/A	N/A			N/A
	User verification and approval (by Admin)				N/A	N/A			N/A
	User access in marketplace		N/A		N/A	N/A			N/A
Service usage	User Login		N/A		N/A	N/A			N/A
	Search in service catalogue		N/A		N/A	N/A			N/A
	Find the		N/A		N/A	N/A			N/A

Steps								
requested service								
Subscription selection		N/A		N/A	N/A			N/A
Sign the contract				N/A	N/A			N/A
Acceptance of Terms & Conditions				N/A	N/A			N/A
Service end2 end usage		N/A		N/A				N/A
Compensation Billing		N/A		N/A	N/A			N/A
Transaction Logs / SLA Management		N/A		N/A				N/A

Table 12: Marketplace Validation Based on QoE Evaluation Parameters

Marketplace Evaluation					
		Quality of Experience			
Process	Steps	Usability (Score 10-100)	UI Design (Score 10-100)	Operations Completeness (Score 10-100)	Service Excellence (Score 10-100)
Service on-boarding	Service files (code) uploading			N/A	N/A
	Service test data uploading			N/A	N/A
	Smart Contract formatting			N/A	N/A
	Terms & Conditions formatting			N/A	N/A
	Service on-boarding completion and testing			N/A	N/A
	Service added in marketplace catalogue				
Data on-boarding	Service selection			N/A	N/A
	Datasets uploading			N/A	N/A
	Smart Contract formatting			N/A	N/A
	Terms & Conditions formatting			N/A	N/A
	Datasets on-boarding completion and testing				
ion in the mark etpla	Link of relevant data / metadata to the service			N/A	N/A

	Launched in marketplace				N/A	
	New service availability announcement					N/A
User enrolment	Company enrolment process				N/A	N/A
	User verification and approval (by Admin)				N/A	N/A
	User (per company) enrolment process (multiple profiles				N/A	N/A
	User type selection (service provider / buyer)				N/A	N/A
	Acceptance of Terms & Conditions				N/A	N/A
	User verification and approval (by Admin)				N/A	N/A
	User access in marketplace					N/A
Service usage	User Login				N/A	N/A
	Search in service catalogue				N/A	
	Find the requested service				N/A	
	Subscription selection				N/A	
	Sign the contract				N/A	
	Acceptance of Terms & Conditions				N/A	
	Service end2 end usage					

7   Conclusions and Next Actions

This deliverable is under the "Testing framework and benchmarking" Task 2.3 of WP2. Through this report the base for overall TRUSTS data marketplace evaluation has been defined in alignment with the scope of the WP2 as per the Grand Agreement.

A presentation of the test case validation toolset is provided along with a description and analysis on the methodologies processes and dependencies with respect to the business validation and technological validation. The testing and validation process life cycle based on agile way of working is also presented.

Information on the TRUSTS data marketplace business validation is analysed given attention to the Lean Startup methodology. Business validation templates has been provided as well as a reference has been given regarding the KPI's as a business validation method for the three-business oriented UC's.

Furthermore, an emphasis was given to the QoE and the QoS as those will be validated via the SUS validation methodology, in the framework of commercial value testing and user's experience.

Moreover, information on the TRUSTS data marketplace with respect to technological validation is presented through this deliverable, focusing on TDD methodology and the setup of testing and user acceptance tests procedures along with templates to be used throughout the lifecycle of this project.

This deliverable constitutes the first version of the two reports defining the methodology and toolset for a comprehensive and robust analysis of the data marketplace technologies and the vertical UC's of the TRUSTS project.

During the whole implementation of the project T2.3 will continue to interact with:

- WP3, which will contribute to the TRUSTS Platform implementation
- WP4, which will provide Privacy Preserving Technologies
- WP5 which will proceed with the execution of the UC trials in order to provide a valuable feedback
- WP7 which will contribute to the adequate business models, to the exploitation and innovation impact assurance

The aim is to systematically assess the input from all involved parties in order to fulfil the objective of T2.3, by validating the three UC's business wise and develop business plans with the highest commercial potential.

This work will be comprehensively analyzed via the deliverable D2.5 entitled "Methodologies for the technological/business validation of use case results II" which is due on M24.