Information and Communication Technologies (ICT) Programme

Project Nº: H2020-ICT-2016-1-732105

D2.2: CERBERO Technical Requirements (Final version)

Lead Beneficiary:	IBM
Workpackage:	WP2
Date:	
Distribution - Confidentiality:	[Public]

Abstract:

This document is meant to describe the intermediate set of technical requirements that have been elicited, according to the project goals and the scenario needs, for the CERBERO framework including methodology to elicit these requirements.

Disclaimer

The CERBERO Consortium is the following:

Num.	Beneficiary name	Acronym	Country
1 (Coord.)	IBM Israel – Science and Technology LTD	IBM	IL
2	Università degli Studi di Sassari	UniSS	IT
3	Thales Alenia Space Espana, SA	TASE	ES
4	Università degli Studi di Cagliari	UniCA	IT
5	Institut National des Sciences Appliquees de Rennes	INSA	FR
6	Universidad Politécnica de Madrid	UPM	ES
7	Università della Svizzera Italiana	USI	CH
8	Abinsula SRL	AI	IT
9	AmbieSense LTD	AS	UK
10	Nederlandse Organisatie Voor Toegepast Natuurwetenschappelijk Onderzoek TNO	TNO	NL
11	Science and Technology	S&T	NL
12	Centro Ricerche FIAT	CRF	IT

For the CERBERO Consortium, please see the http://cerbero-h2020.eu web-site.

Document Authors

The following list of authors reflects the major contribution to the writing of the document.

Name(s)	Organization Acronym

Michael Masin	IBM

The list of authors does not imply any claim of ownership on the Intellectual Properties described in this document. The authors and the publishers make no expressed or implied warranty of any kind and assume no responsibilities for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information contained in this document.

Document Revision History

Date	**Ver.**	**Contributor (Beneficiary)**	**Summary of main changes**
13/02/2019	v1	IBM	Initial draft based on D2.7

Table of contents

1. Executive Summary

Technical Requirements provide a “black box” conceptualization of the target project results with explicit verification tests. The goal of Technical Requirements Elicitation is to ensure that all needs of involved stakeholders are being identified and adequately addressed without prescribing how to achieve them. Whilst elicitation methodology in product or service development is well known, requirements elicitation in large research and innovation projects turns far too commonly into an ad-hoc process carried out without the support of a common, solid methodology. The objective of this deliverable is twofold: to propose a new methodology for the elicitation of technical requirements in research projects in general and to describe the technical requirements of CERBERO.

Firstly, we discuss and highlight where and why this process in large research projects can be significantly different from the one used for product development. Next, we present our methodology based on the best practice for product development and modified to fulfil the research needs. Finally, we apply our elicitation methodology to the CERBERO project.

The methodology suggests starting requirements elicitation process by consolidation of use case requirements and merging them with project operational objectives. Afterwards, the they are enriched with needs of other stakeholders. As a result, 30 CERBERO user needs have been identified. Finally, these user requirements lead to 20 CERBERO technical requirements that, as we verified, cover all identified user needs. These requirements have been traced further on in CERBERO technical deliverables.

A short paper based on the previous version of the deliverable has been accepted to the Requirements Engineering track of 34th ACM/SIGAPP Symposium On Applied Computing (SAC’19) [15].

1.1.Structure of Document

This deliverable is a minor update of D2.7 where the second version of our requirements elicitation methodology and CERBERO requirements have been described. In Section 2 we provide motivation for the new methodology. In Section 3 methodology for elicitation of technical requirements in large research and innovation projects is developed and then, in Section 4, implemented to CERBERO. Finally, Section 5 summarizes the document.

Changes from D2.7:

- In Figure 1. Technical Requirements elicitation process an additional (final) step is added: “Trace of Technical Requirements to WPs/Tasks/Activities”.
- Minor changes in Table 2. CERBERO Use Case Needs and High Level Requirements reflecting minor changes in PE use case without any changes in Technical Requirements and their coverage of user needs.
- Additional references [13]-[15].
- Few typos have been removed.

1.2.Related Documents

CERBERO-D2.1_CERBERO_Scenarios_Description.docx

CERBERO_D2.7_TechnicalSpecification.docx

2. Motivation

According to [1], a requirement is: 1) A need perceived by a stakeholder. 2) A capability or property that a system shall have. 3) A documented representation of a need, capability or property. Technical Requirements (TRs) provide a “black box” definition of project results with explicit verification tests. The Technical Requirements Elicitation (TRE) process should guarantee that all needs of all project stakeholders are considered and adequately addressed and verified, without prescribing how to implement them.

The process for elicitation of TRs, both functional and non-functional, in product or service development is well known, standard procedures are available and widely adopted [2-7]. The same does not apply to large research and innovation projects where TRE turns far too common into an ad-hoc process carried out without the support of a solid methodology.

While the TRE objective remains the same (addressing all user needs), the detailed process and methodology in large research projects should be different from the one used for product development and we intend to propose a methodology, based on the best practice for product and/or service development, but adjusted to research needs.

The main activities in the requirement development process are: elicitation, analysis, specification and validation [2]. In detail, the elicitation activities include fact-finding, requirements gathering, evaluation and rationalization, privatization, and integration. The common TRE methodologies [2-7] focus on needs of product or service users, e.g. [5] propose the following elicitation techniques: interviews, focus groups facilitated workshops, group creativity techniques, group decision-making techniques, questionnaires and surveys, observations, prototypes, benchmarking, context diagrams, and document analysis. All of them mostly target identification of needs of product (service) users. Collaborative Requirements Elicitation in a European Research Project has been described in [13] focusing on collaborative elicitation, refinement and consolidation of software requirements in a distributed environment with heterogeneous stakeholders. The authors propose a three-stage process and collaboration tools to understand needs of directly involved industrial and academic partners.

However, in large research and innovation projects, such as CERBERO, identification of all stakeholders and their roles is not trivial and sometimes counterintuitive to standard requirements analysis of products and services. For example, the main purpose of use case providers in a research project, despite them being traditionally industrials, it is not only developing specific product or service, but rather to evaluate and provide valuable feedback to the research technical, dissemination and exploitation activities. Therefore, using directly Use Case Scenarios and Requirements as the basis for research project TRs could result in requirements not connected to developed technologies or to objectives of the research sponsor authority. It could easily be a case when, in a short-medium term, using existing solutions for product development could be more cost effective and maintainable than using research assets. Moreover, if, e.g., use case providers have their own methodology for requirements elicitation corresponding to their products and services, their methodology should be integrated in the research project considering needs of all involved stakeholders.

Use case providers, technology providers and integrators are all stakeholders with needs that should be satisfied by each other for a successful project. In a philosophical level, it does not contradict modern TRE methodologies, but not supported by them. It becomes too easy to miss an important stakeholder needs leading to ad-hoc practices to address the problem. Finding typical stakeholders and their needs in a research project and proposing generic elicitation process could significantly improve probability of successful collaborative research and its further exploitation.

3. CERBERO Methodology for Elicitation of Technical Requirements in Large Research Projects

As a preparation step, let us identify an initial list of main stakeholders in typical large research projects. Their needs will be merged into User Needs and High Level Requirements and traced to the derived TRs. The TRs will be later traced to goals and activities of technical work packages for appropriate verification and validation tests. We identified the following stakeholders:

- research sponsor, technical reviewer, financial reviewer,
- use case provider with the following roles: 1) technical staff, usually engineers from the providers organization; 2) user; 3) manager,
- research community, industry community, standardization body,
- technology provider and integrator.

Research sponsor's needs are usually represented by call's expected impact. Project proposal translates the expected impact into Project Operational Objectives (POO) adjusting them to the proposed research and innovation. If some expected impact is not mapped to POO (and the project has accepted the sponsor's support), they will become optional User Requirements. We propose using POO as a part of User Requirements instead of mapped part of calls expected impact. POO, in turn, are mapped to project use cases. Again, only partial mapping may occur (especially, for dissemination objectives). The mapped objectives will be traced to different use case needs with use case demonstrations as their validation tests.

Understanding needs of use case users is more complex. Each use case could have very diverse users and each use case provider could have different methodology for use case implementation. We propose a two-step process: first, use case providers define their use case requirements based on use case user needs, then, in the second step, these requirements will be abstracted and generalized to meet projects' level of abstraction and focus. In the first step, use case providers should identify needs of all use case users, e.g., using focus groups, influencing functional context of use cases and, consequentially, use case requirements. The internal methodology could differ very much from one use case provider to another because of company policies and other constraints. The abstraction step is usually done by the whole consortium and it should adjust use case and research levels of abstraction and make the use case be representative for a wide class of industrial use cases.

Research sponsor's needs and use case user needs vary a lot from call to call and from use case to use case. Needs of other stakeholders are common for most research projects and may require only slight adjustment/extension for a specific project. In Table 1 we provide an initial list of needs of all other stakeholders. These needs could apply to most large research projects. Project Advisory Board can help to have on one hand wider perspective and, on another hand, additional project specific needs of relevant industry communities and standardization bodies.

Figure 1 describes the proposed process for TRE. The process starts with elicitation of Use Case Needs and High Level Requirements for all project use cases based on use case scenarios and demonstrations. At this point, different TRE methodologies can be used. Next, the requirements from all use cases should be abstracted and generalized according to project focus and required level of abstraction, e.g., applying [13]. All POO should be either traced to the Use Case Requirements or added to User Requirements list together with unmapped call's expected impacts. We call the set of Use Case Requirements extended of unmapped POO and expected impacts as Core User Requirements. Finally, needs of other stakeholders shown in Table 1

Table 1 would complete the Core User Requirements to the complete list of User Requirements for the project. Based on them, TRs are defined by research consortium with appropriate validation tests tracing between User and Technical Requirements. In the final step, these TR and validation demonstrations should be traced to at least one project work package, task, or activity to ensure addressing the identified needs.

TRE is a complex process, considered by many as art more than engineering. Therefore, our proposal does not guarantee generation of good requirements, but increases probability of considering needs of most (hopefully, all) project stakeholders and, eventually, to project's activities.

Table 1. Common stakeholders and their needs

Stake-holder	Need	Rationale
Technical reviewer (TR)	TR1. View collaborative executable plan TR2. View intermediate results TR3. View technical risks	Strong and effective communication is challenging in large research projects with many partners. Intermediate results and evaluation of technical risks provide valuable feedback for project direction and required mitigation activities.
Financial reviewer (FR)	FR1. View plan vs actual effort FR2. View financial risks	Financial analysis provides important evidence of project status.
Use case technical staff (UCS)	UCS1. Quality of technical results UCS2. Usability of tools UCS3. Technical education UCS4. Technical support	Use case technical staff should apply developed technologies for their product or service in combination with state-of-the-art technologies, and relevant industry standards for better technology evaluation
Use case manager (UCM)	UCM1. Technology cost UCM2. Technology value (improved product quality, staff productivity, reduced time to market, etc.) UCM3. Technology maintainability	Evaluation of business aspects of the technology developed in the project, and risk mitigation.
Research community (RC)	RC1. Timely research publications RC2. Open access to as much tools and data as possible	Repeatability, cross-verification, and reuse of technical results.
Industry community (IC)	IC1. Dissemination of results in all relevant industry communities IC2. Technical education	Dissemination and exploitation of technical results.
Standardization body (SB)	SB1. Contribution to relevant standards	Interoperability of technical results.

Technology Provider (TP)	TP1. Ensure products will use state-of-the-art technology. TP2. Ensure product will support up-to-date standards. TP3. Document the developed technology/ component for product integration	The technology provider will need to stimulate usage of product in research and use feedback to improve product. The technology provider will need to monitor development of standards in research projects and adapt new standards at the appropriate moment (e.g. to compete with competitors and/or to ensure interworking with other product).
Integrator (I)	I1. Use the latest integration methodologies and tools compatible with the developed technology.	The integrator will need to use the latest integration methodologies and tools in order to maintain competitiveness in the market.

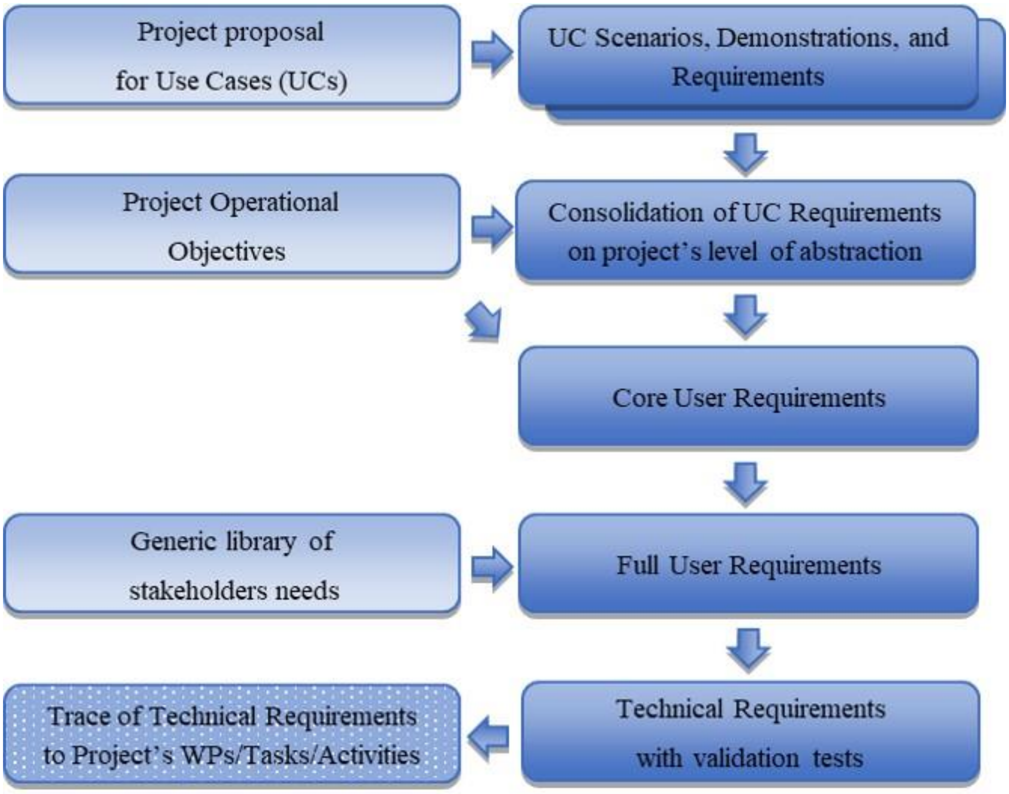

Figure 1. Technical Requirements elicitation process

In the following sections we apply this methodology to CERBERO. CERBERO implementation is based on an iterative approach: the current second version of Technical Requirements will be further updated along the project timeframe. Therefore, the described process that embodies the activities of WP2 will be repeated along the time and updated in the next versions of the report.

4. CERBERO Technical Requirements

In this section we summarize the CERBERO approach and apply all TRE steps to CERBERO.

4.1. Summary of CERBERO Technical Approach and Use Cases

Cyber-Physical Systems (CPS) are complex systems composed of different interacting computing and physical entities that contribute concurrently to determine the behavior of the system as a whole. Computing layer and physical environment are tightly bound; therefore, such systems need to adapt, prospectively and autonomously, to rapid changes in the environment and in the system itself.

CERBERO aims at developing a design environment for component-based CPS based on three pillars: 1) a cross-layer model-based approach to describe, optimize, and analyze the system and all its different views concurrently, 2) an advanced adaptivity support based on a multi-layer self-adaptation strategy, and 3) continuous design environment supporting interoperability crossing tools, multiple layers and different levels of abstraction.

Intrinsic dynamic nature of CPS requires flexibility. While deeply studied, there is no standard solution yet for adaptation and reconfiguration [7]. In particular, *self-reconfiguration and adaptation* have been acknowledged as key features for CPS operators, but existing design frameworks rarely address them. CERBERO Project is meant to address this gap.

Moreover, despite their big promise (considering the claimed enhancement of and the declared speed-up), the existing model-based frameworks are not as popular as it could be expected. Modeling, maintenance, and interoperability overhead, especially with heterogeneous models over several levels of abstraction, are, in fact, challenges not addressed in a satisfactory way [9, 14]. Correctness-by-construction of reusable components implies efficient management of multi-physics, multi-abstraction and multi-fidelity heterogeneous models, capturing and optimizing cross-domain interactions. CERBERO intends to provide such support, designing all the needed components to enable seamless design and operation cycles.

To focus CERBERO effort and evaluate the proposed framework and developed tools, CERBERO defined three use cases, targeting development of CPS in very different levels of abstraction and covering a wide spectrum of system features (going from more hardware-oriented implementations towards more software-oriented ones).

- The *Space Exploration* use case provides self-monitoring and self-healing capabilities by means of continuous monitoring- and/or processing-based system reconfiguration techniques to overcome the failures caused by the radiation or the harsh environmental conditions. In this use case CERBERO framework should open a new opportunity for hardware/software co-design of robotic arms by using commercial of-the-shelf (COTS) reconfigurable FPGA to address most common space requirements, instead of expensive specialized FPGAs currently used.

- The *Smart Travelling for Electric Vehicle* use case focuses on the assistance an electric vehicle can give to the driver, when confronted with the task of driving the car from a given origin place A to a given destination B. This assistance must also assure that sufficient battery capacity is available to complete the route. The *Smart Travelling for Electric Vehicle* use case challenges system level design of universal simulation for complete driver experience, where CERBERO framework should provide a reconfigurable solution for driver support interface for different car types, driving scenarios, drivers profile and changing physical context of both the environment and car itself, integrating with an already existing complex simulation environment [10].

- The *Ocean Monitoring* use case comprises smart video-sensing unmanned vehicles with immersive environmental monitoring capabilities. They serve as marine eyeballs that can capture live videos and images of the local on-sea and subsea surroundings [11, 12]. The *Ocean Monitoring* use case starts from system level and eventually combines system and HW/SW co-design levels for development of underwater ocean monitoring robots. These robots may operate both in guided modes from the shore and autonomously for a large variety of monitoring and navigating tasks in changing environmental conditions.

4.2. Consolidation of CERBERO Use Case Needs and High Level Requirements

Project use cases are usually developed using use case provider's methodology that will change from partner to partner. At M9 in [D2.3] all the different requirements coming from all project stakeholders have been identified with different levels of maturity. These requirements where updated and traced to User Needs and High Level Requirements in [D2.4]. Table 2 summarizes them with proposed demonstrations for their validation.

Table 2. CERBERO Use Case Needs and High Level Requirements

Use Case	Requirement	Validation demonstration
Self-Healing System for Planetary Exploration (PE)	PE1. Enable Dependable Hardware / Software (HW/SW) co-design. **Need**: minimization of energy consumption and costs, while keeping/improving resiliency.	Multi-objective Architecture design of arm using COTS FPGA and considering life-cycle costs, energy efficiency, reliability, etc. Trajectory generation and status monitoring applications.
	PE2. Develop integrated open-source or commercially available toolchain environment for the design and assessment of heterogeneous cyber-physical systems. **Need**: provide multi-objective design space exploration and multi-view analysis; reduce development time of complex heterogeneous systems by increasing the level of abstraction; increase quality and verification level to ensure proper operation of the system.	Software and System in-the-loop simulation based on high-level applications abstractions. Interoperability between HW/SW co-design tools on different levels of abstraction
	PE3. Development of a (self-) adaptation methodology and supporting tools. **Need**: efficient support of architectural adaptivity, according to radiation effects and harsh environmental conditions.	Self-healing and run time adaptation features.
Smart Travelling for Electric Vehicles (ST)	ST1. Development of parametric, modular and extendable cyber-physical co-simulation environment. **Need**: reduction of costs, increase of re-use in different simulation scenarios	Modular communication protocols and time synchronization. Logging application. Building Battery and Motor modules from generic components.

		Modular and extendable Driver support module. Safe, Secure, and Private Adaptive routing module with energy and cost efficiency and sensitive to drivers needs and environmental status.
	ST2. Development of an integrated open-source or commercially available tool-chain for design space exploration and co-simulation, with system-in-the-loop capabilities. **Need**: reduce development, verification, and integration time and costs by a library of reusable components and metrics integrated by common framework in different levels of abstraction; incremental prototyping.	Software in-the-loop simulation. Interoperability of System Level Design tools.
	ST3. Development of a (self-)adaptation methodology with supporting tools. **Need**: efficient support of functional adaptivity, according to system, human and environment triggers.	Re-routing in different simulation scenarios
Ocean Monitoring robot (OM)	OM1. Provide complete design cycle from system level design to HW/SW co-design and implementation of Ocean Monitoring robot using adaptable COTS. **Need**: reduction of energy consumption and costs, increase reuse in other projects, while keeping or improving safety and security level and maintenance costs.	Development of Adaptive Camera based on COTS HW with OEM firmware. Data storage according to mission needs On demand task dependent Data Fusion. Secure communication. Building Battery and Motor modules from generic components.
	OM2. Develop integrated “open” tool-chain environment for development of Ocean Monitoring robots with focus on incremental prototyping. **Need**: facilitate development cycles, reduce time to market, and increase reuse, quality and verification level by incremental prototyping from high level of abstraction directly to working real time applications.	Incremental prototyping of Adaptive Camera components from high level models.
	OM3. Development of a (self-)adaptation methodology with supporting tools.	Develop adaptive image enhancement methods for Adaptive Camera.

	Need: Efficient support of functional adaptivity, according to system, human and environment triggers.	Multi-objective navigation and motor control modules with run time adaptation for Autopilot.

4.3. Mapping of CERBERO Operational Objectives to Use Case Needs and Demonstrations

In research projects it is usually possible to trace some Project Operational Objectives (POO) to Use Case Requirements and validate by use case demonstrations. Table 3 shows how it has been done in CERBERO.

Table 3 – Mapping of CERBERO Operational Objectives to Use Case Requirements

	Objective	UC High Level Requirement	Validation demonstration
CH1.1	Provide reusable Libraries of Key Performance Indicators (KPIs), Cross-Layer Models and Adaptivity support.	PE1,3, ST1,3, OM1,3	PE: Architecture, run time adaptation, trajectory generation, self-healing, reliability, and dependability ST: Modular simulation with timing, energy, reliability, safety and security related KPIs OM: Image quality, response time, power, energy, and throughput related KPIs
		PE1,3, ST1,3, OM1,3	PE: Architecture, run time adaptation, trajectory generation, self-healing ST: system-in-the-loop, battery and motor models including non-functional concerns, reliable, safe and secure driver support module OM: Components, adaptive camera systems, secure wireless communication, and marine navigation
CH1.2	Provide a comprehensive framework, customizable upon the UC needs, extending and making interoperable a large set of tools.	PE2, ST2, OM2	PE: All ST: All OM: All

CH1.3	Reduce by 30% the energy consumed by a fully CERBERO compliant CPS or CPSoS, while maintaining its performance.	PE1,3, ST1,3, OM1,3	PE: Architecture, run time adaptation, trajectory generation ST: Driver Support module, battery module, motor module, system in the loop functionality OM: Adaptive Camera systems, marine navigation, propulsion/motor control, physical design
CH2.1	Reduce DSE by an order of magnitude.	PE1, ST1, OM1 PE2, ST2, OM2	PE: Self-healing, scalability, architecture ST: system-in-the loop functionality, driver support OM: incremental prototyping
CH2.2	Reduce by 50% the design efforts required to build a CPS of a given performance.	PE2,3, ST2,3, OM2,3	PE: Run time adaptation ST: All OM: sub-optimal hull, propulsion, or battery solution can lead to a deer design. For example, an engine can cost \$10,000 to \$30,000, battery solution between \$3,000 to \$15,000 to gain speed and range.
CH2.3	Reduce by 50% cost of maintenance.	PE1,3, ST1,3, OM1,3	PE: Self-monitoring, self-healing, scalability ST: System in the loop, driver support OM: multi-objective design, reduced number of physical components, COTS

4.4. Full List of CERBERO User Needs and High Level Requirements

In this section we combine unmapped Operational Objectives, Use Case Needs and the needs of other stakeholders defined in Table 1.

1. OO1. Plan CERBERO results in at least 3 industrial products.
2. OO2. Provide a fully marketable version of the CERBERO modelling and design environment.
3. OO3. Foster Interoperability.
4. PE1. Enable Dependable Hardware / Software (HW/SW) co-design.
5. PE2. Develop integrated open-source or commercially available toolchain environment for the design and assessment of heterogeneous cyber-physical systems.
6. ST1. Develop reconfigurable extendable modular simulation environment for smart travelling driver interfaces.
7. ST2. Develop integrated "open" toolchain environment for development of simulation modules and their integration with focus on modular integration with existing virtual environment.
8. OM1. Provide complete design cycle from system level design to HW/SW co-design and implementation of Ocean Monitoring robot using adaptable COTS.
9. OM2. Develop integrated "open" toolchain environment for development of Ocean Monitoring robots with focus on incremental prototyping.
10. PE3/ST3/OM3. Development of a (self-)adaptation methodology with supporting tools.

11. TR1. View collaborative executable plan.
12. TR2. View intermediate results.
13. TR3. View technical risks.
14. FR1. View plan vs actual effort.
15. FR2. View financial risks.
16. UCS1. Quality of technical results.
17. UCS2. Usability of tools.
18. UCS3. Technical education.
19. UCS4. Technical support.
20. UCM1. Technology cost.
21. UCM2. Technology value (improved product quality, staff productivity, reduced time to market, etc.).
22. UCM3. Technology maintainability.
23. RC1. Timely research publications.
24. RC2. Open access to as much tools and data as possible.
25. IC1. Dissemination of results in all relevant industry communities.
26. IC2. Technical education.
27. SB1. Contribution to relevant standards.
28. TP1. Ensure products will use state-of-the-art technology.
29. TP2. Ensure product will support up-to-date standards.
30. I1. Support the latest integration methodologies and tools.

4.5. CERBERO Technical Requirements

In this section we define CERBERO technical requirements and verify their sufficiency in the next section. When the phrases MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, or OPTIONAL are being used, these have the following meaning and interpretations:

- **MUST, REQUIRED, SHALL** - means an absolute requirement for the specification.
- **MUST NOT, SHALL NOT** - means an absolute prohibition for the specification.
- **SHOULD, RECOMMENDED -** valid reasons may exist in certain cases to avoid the requirement. However, its full implication must be understood and carefully considered before choosing an alternative direction/option.
- **SHOULD NOT, NOT RECOMMENDED** - valid reasons may exist when the requirement could be acceptable or even useful. However, its full implications should have been carefully considered understood and prior to implementing anything in breach of this.
- **MAY, OPTIONAL -** simply means that the requirement is truly optional, nice to have. An implementation that does not fulfil an optional requirement MUST be prepared to function together with another implementation that fulfils/implements this option, and vice versa.

Technical Requirement	**Trace to WPs**
CERBERO-0001. CERBERO framework/technology SHOULD increase the level of abstraction at least by one for HW/SW co-design and for System Level Design.	3, 4, 5
CERBERO-0002. CERBERO framework/technology SHOULD provide interoperability between cross-layer tools and semantics at the same level of abstraction.	3, 5
CERBERO-0003. CERBERO framework/technology SHOULD provide incremental prototyping capabilities for HW/SW co-design.	4, 5

CERBERO-0004. CERBERO framework/technology SHOULD provide software and system in-the-loop simulation capabilities for HW/SW co-design and System Level Design.	3, 5
CERBERO-0005. CERBERO framework/technology SHOULD provide multi-viewpoint multi-objective correct-by-construction high-level architecture.	3, 5
CERBERO-0006. CERBERO framework/technology SHOULD ensure energy efficient and dependable HW/SW co-design using cross-layer run time adaptation of reconfigurable HW.	3, 4, 5
CERBERO-0007. CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable functional and non-functional KPIs.	3
CERBERO-0008. CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable energy aware computing infrastructure.	3, 4
CERBERO-0009. CERBERO SHALL develop integration methodology and framework.	3, 4, 5
CERBERO-0010. CERBERO SHALL provide Open Data Management Plan	1
CERBERO-0011. CERBERO SHALL have Dissemination and Exploitation Plans • in relevant industry communities • for technical education • for standardization effort	7, 8
CERBERO-0012. CERBERO Exploitation Plan SHALL • consider at least 3 industrial products, • have a business model for the integration methodology and framework.	8
CERBERO-0013. All CERBERO APIs between tools and most of CERBERO tools SHOULD have open source licence.	5, 8
CERBERO-0014. CERBERO WP and task leaders SHALL organize scheduled face to face and remote meetings.	All
CERBERO-0015. CERBERO SHALL provide review reports including • intermediate results, • technical risks evaluation, • plan vs actual effort, • financial risks evaluation.	All
CERBERO-0016. CERBERO tools/technologies SHOULD be tested vs state-of-the-art	3, 4, 5
CERBERO-0017. CERBERO Use Case providers SHOULD check and provide timely feedback on the usability of CERBERO tools and framework.	6
CERBERO-0018. CERBERO technology providers SHALL prepare face to face or online tutorials / education for use case engineers.	3, 4, 5
CERBERO-0019. CERBERO technology providers SHALL coordinate technical support and documentation for their tools with use case engineers.	3, 4, 5
CERBERO-0020. CERBERO framework SHALL provide methodology and tools for development of adaptive applications.	3, 4, 5

4.6. Validation of User Needs

The following table validates sufficiency of Technical Requirements described above.

Table 4. Mapping User Requirements to Technical Requirements and Validation Tests

User Requirement	Technical Requirement(s)	Validation Test

OO1. Plan CERBERO results in at least 3 industrial products.	**12** (Exploitation Plan MUST consider at least 3 industrial products.)	Exploitation Report meets Plan for CERBERO contribution to at least 3 industrial products.
OO2. Provide a fully marketable version of the CERBERO modelling and design environment.	**12** (Exploitation Plan SHALL have a business model for the integration methodology and framework.)	Exploitation Report meets Plan for the integration methodology and framework.
OO3. Foster Interoperability.	**13** (CERBERO SHOULD provide all API between tools and most of tools with open source license.) **9** (CERBERO SHALL develop integration methodology and framework.)	Demos with multiple tools, both cross-layer and from different levels of abstraction (All).
PE1. Enable Hardware / Software (HW/SW) co-design for Rad-Tolerant control of robotic arm for planetary exploration using adaptable COTS FPGAs.	**5** (CERBERO framework/technology SHOULD provide multi-viewpoint multi-objective correct-by-construction high-level architecture.) **6** (CERBERO framework/technology SHOULD ensure energy efficient and dependable HW/SW co-design using cross-layer run time adaptation of reconfigurable HW.) **7** (CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable functional and non-functional KPIs.) **8** (CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable energy related components.)	PE demonstrations
PE2. Develop integrated “open” toolchain environment for development of robotic arms for space missions with focus on multi-viewpoint system-in-the-loop virtual environment.	**1** (CERBERO framework/technology SHOULD increase the level of abstraction at least by one for HW/SW co-design and for System Level Design. **2** (CERBERO framework/technology SHOULD provide interoperability between cross-layer tools and semantics at the same level of abstraction.) **4** (CERBERO framework/technology SHOULD provide software and system in-the-loop simulation capabilities for HW/SW co-design.)	PE demonstrations
ST1. Develop reconfigurable extendable modular simulation environment for smart travelling driver interfaces.	**5** (CERBERO framework/technology SHOULD provide multi-viewpoint multi-objective correct-by-construction high-level architecture.) **7** (CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable functional and non-functional KPIs.)	ST demonstrations

	8 (CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable energy related components.)	
ST2. Develop integrated “open” toolchain environment for development of simulation modules and their integration with focus on modular integration with existing virtual environment.	**2** (CERBERO framework/technology SHOULD provide interoperability between cross-layer tools and semantics at the same level of abstraction.) **4** (CERBERO framework/technology SHOULD provide software and system in-the-loop simulation capabilities for System Level Design.)	ST demonstrations
OM1. Provide complete design cycle from system level design to HW/SW co-design and implementation of Ocean Monitoring robot using adaptable COTS.	**5** (CERBERO framework/technology SHOULD provide multi-viewpoint multi-objective correct-by-construction high-level architecture.) **6** (CERBERO framework/technology SHOULD ensure energy efficient and dependable HW/SW co-design using cross-layer run time adaptation of reconfigurable HW.)	OM demonstrations
OM2. Develop integrated “open” toolchain environment for development of Ocean Monitoring robots with focus on incremental prototyping.	**1** (CERBERO framework/technology SHOULD increase the level of abstraction at least by one for HW/SW co-design and for System Level Design.) **2** (CERBERO framework/technology SHOULD provide interoperability between cross-layer tools and semantics at the same level of abstraction.) **3** (CERBERO framework/technology SHOULD provide incremental prototyping capabilities for HW/SW co-design.) **7** (CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable functional and non-functional KPIs.) **8** (CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable energy related components.)	OM demonstrations
PE3/ST3/OM3. Development of a (self-)adaptation methodology with supporting tools.	**6** (CERBERO framework/technology SHOULD ensure energy efficient and dependable HW/SW co-design using cross-layer run time adaptation of reconfigurable HW.) **20** (CERBERO framework/technology SHALL provide methodology and tools for development of adaptive applications.)	All UC demonstrations with reconfigurability.

TR1. View collaborative executable plan.	**14** (CERBERO WP and task leaders SHALL organize scheduled face to face and remote meetings.)	Review reports with scheduled face to face and remote meetings.
TR2. View intermediate results.	**15** (CERBERO SHALL provide review reports with intermediate results.)	Periodic review and demos at GA meetings
TR3. View technical risks.	**15** (CERBERO SHALL provide periodic reports with technical risks evaluation.)	Periodic status meetings / calls with agenda and minutes
FR1. View plan vs actual effort.	**15** (CERBERO SHALL provide review reports with plan vs actual effort.)	Periodic review
FR2. View financial risks.	**15** (CERBERO SHALL provide periodic reports with financial risks evaluation.)	Periodic review
UCS1. Quality of technical results	**16** (CERBERO tools SHOULD be tested vs state-of-the-art.)	CERBERO deliverables and peer reviewed publications
UCS2. Usability of tools	**17** (CERBERO Use Case providers SHOULD check and provide timely feedback on the usability of CERBERO tools and framework.)	Periodic report and questionnaires
UCS3. Technical education	**18** (CERBERO technology providers SHALL prepare face to face or online tutorials / education for use case engineers.)	Tool's usage in use case
UCS4. Technical support	**19** (CERBERO technology providers SHALL coordinate technical support for their tools with use case engineers.)	Tool's usage in use case
UCM1. Technology cost	**13** (All CERBERO API and most of CERBERO tools SHOULD have open source licence.)	All APIs between tools and at least 80% of tools have open source licence that permit free commercial use.
UCM2. Technology value (improved product quality, staff productivity, reduced time to market, etc.)	**1** (CERBERO framework/technology SHOULD increase the level of abstraction at least by one for HW/SW co-design and for System Level Design.) **2** (CERBERO framework/technology SHOULD provide interoperability between cross-layer tools and semantics at the same level of abstraction.) **3** (CERBERO framework/technology SHOULD provide incremental prototyping capabilities for HW/SW co-design.) **4** (CERBERO framework/technology SHOULD provide software and system in-the-loop simulation capabilities for HW/SW co-design and System Level Design.) **5** (CERBERO framework/technology SHOULD provide multi-viewpoint	Business value of all demonstration in all use cases

	multi-objective correct-by-construction high-level architecture.) **6** (CERBERO framework/technology SHOULD ensure energy efficient and dependable HW/SW co-design using cross-layer run time adaptation of reconfigurable HW.) **7** (CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable functional and non-functional KPIs.) **8** (CERBERO framework/technology SHALL define methodology and SHOULD provide library of reusable energy related components.)	
UCM3. Technology maintainability	**12** (Exploitation Plan SHALL have a business model for the integration methodology and framework.)	Exploitation Report meets Plan for the integration methodology and framework.
RC1. Timely research publications	**11** (Dissemination Plan.)	Dissemination Report meets Plan
RC2. Open access to as much tools and data as possible	**13** (All CERBERO API between tools and most of CERBERO tools SHOULD have open source licence.) **10** (Open Data Management Plan)	All API and at least 80% of tools have open source licence that permit free commercial use. Open Data storage is established according to the plan.
IC1. Dissemination of results in all relevant industry communities	**11** (CERBERO MUST have Dissemination and Exploitation Plans in relevant industry communities.)	Dissemination and Exploitation Reports meet Plans in relevant industry communities
IC2. Technical education	**11** (CERBERO MUST have Dissemination and Exploitation Plans for technical education.)	Dissemination and Exploitation Reports meet Plans standardization effort
SB1. Contribution to relevant standards	**11** (CERBERO MUST have Dissemination and Exploitation Plans for standardization effort.)	Dissemination and Exploitation Reports meet Plans standardization effort
TP1. Ensure products will use state-of-the-art technology.	**16** (CERBERO tools SHOULD be tested vs state-of-the-art.)	CERBERO deliverables and use case demonstrations. Peer reviewed publications on tools and developed technologies.

TP2. Ensure product will support up-to-date standards.	**11** (CERBERO SHALL have Dissemination and Exploitation Plans for standardization effort)	Dissemination and Exploitation Reports meet Plans standardization effort.
I1. Support the latest integration methodologies and tools.	**9** (CERBERO SHALL develop integration methodology and framework.) **16** (CERBERO tools SHOULD be tested vs state-of-the-art)	All use case demonstrations. Peer reviewed publication on integration methodology and framework.

Based on the technical requirements and demonstrations, tools and technology providers will perform gap analysis and define activities in technical WPs to fill the gaps of the framework components (which can be enhanced available tools or new components), integration framework itself, and required model transformations between tools.

5. Conclusion

In this document, CERBERO methodology for Technical Requirements Elicitation has been presented. Moreover, by applying the proposed methodology we identified the Technical Requirements that are driving CERBERO activities.

This document is going to be updated again at M26. In this way we will guarantee an effective industry-driven deployment of the CERBERO framework and technologies.

6. References

[CERBERO 2017]	http://www.cerbero-h2020.eu
[D1.1]	CERBERO_D1 1_KoMprogress_UniSS_FF1_20170512.docx
[D1.6]	CERBERO_D1.6_Open_Data_Management_Plan_TNO_FF1_20171009.docx
[D2.3]	CERBERO_D2.3_ScenariosDescription_TASE_FF1_20171009.docx
[D7.7]	CERBERO_D7.5_DisseminationPlan_USI_FF1_20171009
[D8.3]	CERBERO_D8.3_InnovationStandardisationExploitationPlan_AI_FF1_20171009
[CPS Framework R1.0]	CPS Public Working Group Cyber-Physical Systems (CPS) Framework Release 1.0 (https://pages.nist.gov/cpspwg/)
[1]	IREB, “Glossary of requirements engineering terminology,” 2014.
[2]	G. Kotonya and I. Sommerville, *Requirements Engineering: Processes and Techniques*. plus 0.5em minus 0.4emJohn Wiley and Sons, 1998.
[3]	“Systems and software engineering – system life cycle processes,” ISO/IEC 15288:2008, 2008.
[4]	INCOSE, “Incose systems engineering handbook - a guide for system life cycle processes and activities,” 2011.
[5]	P. M. Institute, *A Guide to the Project Management Body of Knowledge: PMBOK(R) Guide*, 5th ed. plus 0.5em minus 0.4emProject Management Institute, 2013.
[6]	INCOSE, “Guide to the systems engineering body of knowledge (sebok),” 2015.
[7]	NIST. (2016) Framework for cyber-physical systems, release 1.0. [Online]. Available: https://pages.nist.gov/cpspwg/ =0pt
[8]	M. Masin, F. Palumbo, H. Myrhaug, O. J. A. de Filho, M. Pastena, M. Pelcat, L. Raffo, F. Regazzoni, A. A. Sanchez, A. Toffetti, E. de la Torre, and K. Zedda, “Cross-layer design of reconfigurable cyber-physical systems,” in *DATE Conference*, 2017.
[9]	J. Sztipanovits, T. Bapty, S. Neema, X. Koutsoukos, and J. Scott, “The meta toolchain: Accomplishments and open challenges,” no. ISIS-15-102, 2015.
[10]	OKTAL. (2017) Scaner studio. [Online]. Available: http://www.oktal.fr/en/automotive/range-of-simulators/software =0pt
[11]	L. Kaliciak, H. Myrhaug, and A. Goker, “Unified hybrid image retrieval system with continuous relevance feedback,” in *21st World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI)*, 2017.
[12]	L. Kaliciak, H. Myrhaug, and A. Goker, “Content-based image retrieval in augmented reality,” in *ISAmI Conference*, 2017.
[13]	Guglielmo De Angelis, Alessio Ferrari, Stefania Gnesi, and Andrea Polini, “Collaborative Requirements Elicitation in a European Research Project”. *ASM SAC ’16*, 2016
[14]	G Liebel, N Marko, M Tichy, A Leitner, and J. Hansson. 2014, “Assessing the State-of-Practice of Model-Based Engineering in the Embedded Systems Domain”. *MODELS*, 2014
[15]	M. Masin, F. Palumbo, J.Adriaanse, H.Myrhaug, F.Regazzoni, M. Sanchez, and K. Zedda. 2019. Elicitation of Technical Requirements in Large Research Projects: the CERBERO approach. *ACM SAC Conference (SAC’19)*, 2019.