



Emilia Brad<sup>1</sup> and Stelian Brad<sup>1,2,\*</sup>

- <sup>1</sup> Department of Engineering Design and Robotics, Technical University of Cluj-Napoca, Bvd. Muncii 103-105, 400641 Cluj-Napoca, Romania; emilia.brad@muri.utcluj.ro
- <sup>2</sup> Cluj IT Cluster, Memorandumului 28, 400441 Cluj-Napoca, Romania
- \* Correspondence: stelian.brad@staff.utcluj.ro; Tel.: +40-730-017-126

Abstract: A particular characteristic of disruptive products is in reengineering advanced technologies for addressing the needs of low-end consumers and/or non-consumers, to transform them into new consumers. This requires a lean co-creative analysis of requirements with all stakeholders involved. Even if a theory encourages the continuous connection of designers and users throughout the design lifecycle for agile adaptation of requirements to the new experiences of users by intersecting them with various versions of the prototype, the rigid budget and time allocated to the design project require novel approaches to clarify the right vectors of product-evolution from the very early design stages of the project lifecycle—allowing agile approaches to fine-tune the set of requirements. In this context, an analysis process of requirements that uses a constructor inspired by living systems is introduced in this paper. This constructor identifies gaps in requirement formulation and indicates areas where improvements must be undertaken. The method is applied in the case of a new cybersecurity software solution that targets micro and small companies.

**Keywords:** requirements analysis; product planning; disruptive engineering; disruptive product; living system framework

# 1. Introduction

Research on product requirements goes back more than 70 years [1]. Nowadays we can count a significant number of contributions in this area. For example, concerning requirements analysis, there are almost 160,000 papers indexed in Clarivate Analytics from over the last 10 years. This is a strong indicator of the importance of a good and comprehensive definition of product requirements to develop a competitive solution. Competitiveness strongly depends on fulfilling customers' and users' requirements and expectations, but this is mostly related to one projection of the problem; that is, the one dealing with the adoption of a product. There are also projections connected to other stakeholders, such as producers, strategic partners, shareholders, etc. [2]. This reveals a complicated and complex landscape the definition and analysis of requirements. Complicatedness occurs because of many players that are involved, as well as from the large number of angles from which the product must be analyzed [3]. For example, there are different types of requirements some of them are related to functionalities, some others with usability, and some others are focused on quality performance, etc. [4]. Complexity is mainly generated by conflicts that occur between different requirements [5]. For example, there are negative correlations between low product costs and high-quality performance. When the product is designed with a lifecycle perspective in mind, engineers face a large set of target-functions [6,7]. Things become even more challenging when engineers have to design innovative products, with little to no discussion of the current products. In such cases, all categories of stakeholders have difficulties in articulating the proper set of requirements-especially the end-users [8–10]. In these situations, it is also very difficult, if not impossible, to prove that the set of requirements is formulated properly and comprehensively. This is the main



Citation: Brad, E.; Brad, S. Requirements Analysis in Disruptive Engineering Solutions Using the Paradigm of Living Systems. *Appl. Sci.* 2021, *11*, 9854. https://doi.org/ 10.3390/app11219854

Academic Editors: Federico Rotini, Niccolò Becattini, Yuri Borgianni and Lorenzo Fiorineschi

Received: 3 September 2021 Accepted: 20 October 2021 Published: 21 October 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).



reason for which agile approaches for product development have gained interest in the last 10–15 years [11–13]. These approaches, such as agile/SCRUM [14], spiral [15], and lean [16], consider that developers and end-users (as well as other categories of stakeholders) must work together, in sessions of gradual iterations, to formulate and discover what they really want from the new product by combining creativity and background experience with foreground experience over the journey of iterative prototyping of the new product [17–19]. There are some premises necessary to make these approaches effective, too. One of the necessary conditions is to have sufficient time and budget to ensure a multi-faceted exploration of the innovation. Another condition is to have cooperative end-users, properly selected and representative in terms of critical mass. But concerning agile methodologies for requirements definition and analysis, one crucial condition is to relate with end-users that have background experience in the field. This means that agile methodologies can be effective if the new product addresses high-end consumers, because this category of consumers is capable of articulating requirements and expectations about future products with concern to their past and present experiences with former products in the same family, as well as with concern to the vision they can see for the new product.

Nowadays, a special category of consumers is coming to the attention of producers. This category is that of non-consumers. Concerning this category, producers are interested in designing and launching so-called "disruptive innovative products" [20]. Disruptive products, which can be better coined disruptive product-service systems [21], or disruptive solutions, address special niches in the market that have the potential to transform non-consumers into low-end consumers. These market segments have very little to no experience with products from the family of products that the innovation intends to target. It is challenging to define requirements with these end-users because they cannot articulate what the product must fulfill and how to fulfill it. In such cases, end-users must first be educated to understand the value of the new product for them and to formulate the proper product vision in this respect. Requirements are mostly generated by experts indirectly, by extracting observations about end-users' behavior in simulated environments and created contexts (e.g., Go-to-Gemba [22], contextual inquiry [23], job-to-be-done [24], one-to-one interviews [25], persona analysis [26], user journey [27]). However, even these methods are not strong enough to elaborate a valid (i.e., complete and consistent) set of requirements in the early stages of the new product development process because they miss the context for measuring potential gaps.

In this problem-space, the research question that the present paper intends to address is "How can we establish if a complete and consistent set of requirements is in place such that we can move forward with the new product development; thus, being confident that agile or similar approaches will work well, even for disruptive products that target non-consumers?". Confidence is reached when we need agile methodologies only for fine-tuning of the solution, based on the feedback received from the end-users as a result of their interactions with various types and levels of prototypes. Valid requirements enable the extraction and use of the proper set of design approaches (or design processes). These further enable the generation of appropriate outputs (the set of reliable solutions). Validity refers to consistency and completeness concerning the "real world" (i.e., the world projected from the eyes of the end-users, harmonized with their cultural background). Incompleteness occurs when some critical requirements are missing from the list, leaving engineers to make design choices that are not aligned with the end-user expectations [28]. Completeness has both an internal and external dimension; internal completeness refers to the closeness of a requirement specification to the statements and inferences that can be made exclusively based on that specification. External completeness refers to the inclusion of all pertinent information within the requirement's specification.

Consistency (or inconsistency) is in strong correlation with the methods and methodologies that are used in the requirements analysis process. Inconsistency is also caused by requirements that conflict with each other or conflict with the new product policy [28]. For avoiding the formulation of inconsistent requirements (often called "useless requirements"), appropriate tools must be applied for requirement analysis. A requirement is consistent when it comprises clear answers to the following questions: (a) What is the significance of the judged requirement? (b) What is the measuring unit and on which scale is it measured? (c) How will the measurement process be performed to receive trustworthy information? (d) What is the ultimate goal in terms of requirement fulfilment? (e) What must be the minimum level of achievement? (f) To what extent is it going to be achieved within a well-defined time (nominal value and tolerance)? (g) How is the information about evolution to be recorded? (h) What is the level of achievement in other comparable systems and why?

It is thus the purpose of this paper to investigate former research contributions in handling the challenge of defining complete and consistent requirements in the early development stage of disruptive products. Based on the identified gaps from the literature review, this paper continues with a proposal for robustly tackling this problem.

In this respect, the next section of the paper provides a synthesis of previous research for the purpose mentioned above. This investigation is not limited to methods strictly dedicated to a certain category of products (or product-service systems), because the goal is to formulate a new contribution that has a high degree of universality, being applicable for any type of product (e.g., mechanical, electrical, software, mechatronic, or even pure services or product-service systems).

## 2. Related Work

To analyze current research about methods for requirements analysis, we have investigated papers published in the databases of Clarivate Analytics and SCOPUS. The combination of words "requirements analysis [in the topic]" AND " product [in the title]" AND "method [in the title]" listed 89 papers in Clarivate Analytics. With the support of VOSviewer [29], connections between the most representative concepts and issues included in these papers have been analyzed. They show that "uncertainty" is mostly connected to "(customer/product) requirements" and the "way" they are approached, whereas "techniques" influence "quality requirements" and "impact", and they depend on "product line". "Customer" connects "techniques" and "uncertainty". The bulk of aspects treated in the selected papers is on planning, assessment, functionality, and innovation—this was also revealed by VOSviewer. A detailed look within the content of the selected papers reveals that the major focus of the proposed methods is not on analyzing the completeness and consistency of requirements, but rather on how to use them to better apply results to new product designs.

Searches were reconsidered, with a focus on "requirement consistency [in the title]", combined with "requirement analysis [in the topic]". Clarivate Analytics and SCOPUS reported 14 papers, but only 10 of them were related to product design. An important remark is that most of the papers dealing with quality analysis of requirements are older than 10 years. This might be connected with the fact that in the last decade, agile methods and methodologies have gained more attention. However, new contexts can change the game, as is the case with disruptive innovations that target non-consumers, where new methods must be added to enhance the modern practices promoted by agile methodologies [30]. Searches on "requirement completeness [in the title]" from Clarivate Analytics and SCOPUS returned 21 titles, of which 6 papers were focused on methods for assessing the completeness of requirements.

The conclusion is that only very little research has focused on analyzing the consistency and completeness of requirements. Another remark is that the majority of research is focused on software products. Software products are much more complex than other products from the perspective of requirement definition, because of their invisibility (intangibility), complexity, and intensity of effort [30]. Requirement analysis in terms of completeness and consistency is an essential aspect in product innovation—this is either done in the early stages or in any other subsequent stages of product development, and it is independent of the types of methodologies and tools used to collect requirements from stakeholders. Table 1 illustrates the main findings from the literature review.

**Table 1.** A synthetic view of related work.

Developments and Benefits	Limitations	Reference
Software tool for conditional requirements and business rule syntax, user-defined glossaries for parsing documents, parsing and extracting structured content, finding the use of problematic phrases, requirements dependency analysis, identifying systems that interact each other and missing non-functional requirements	Requirements are analyzed in the form of a pseudo-code and high-level language, which represents a barrier to working in a friendly way with end-users (especially in the case of disruptive innovations where we need to work with low-end consumers).	[30]
Method that considers domain models and domain documentation to capture requirements. This involves experts to formulate domain concepts.	It focuses only on external completeness, focuses only on functional requirements, counts a lot on subject-matter experts.	[31]
Software tool to test and debug formal requirements using property-based design flow and simulation of realizable formal specifications (game-based debugging approach for specification)	It requires expertise to formalize requirements. Hard to follow in interactions with end-users.	[32]
Method that employs a combination of topic analysis, persona, and scenario approaches to define requirements in relation to multiple actors.	It does not analyze completeness and consistency of requirements.	[33]
Method to formalize requirements based on AUTOSAR standards applied in the automotive industry. It provides a translation of requirements into logical constraints which enable the use of specific solvers to analyze the consistency of requirements.	Difficult to transfer the method to disruptive products that address low-end consumers, in the sense of using the method in the co-creation process. It also does not provide a framework for completeness analysis of requirements.	[34]
Approach that uses a low-level functional formalism to simplify the analysis process. It breaks down the specification into smaller, analyzable parts and then uses functional composition rules to ensure that verified properties hold for the entire specification.	It requires a very good understanding of the application domain and applies for well-established products. It is not effective in the case of exploring new products and for interviewers that have little or no understanding of the application domain. It can be applied only to requirements that are already	[35]
Technique for automated consistency analysis of requirements that are formalized based on patterns. It reduces the analysis time of consistency.	Difficult to apply the Satisfiability Modulo Theory solvers of this method with the end-users from the category envisaged by disruptive products dedicated to low end-consumers.	[36]
A methodology that analyzes requirement consistency based on semantics, using UML formats.	It can work well with experts in UML but not with outsiders, such as end-users with no previous expertise and experience in the application domain.	[37]
A technique for automatic analysis of consistency between software requirements and detailed designs, achieved at the expense of using imprecise data-flow analysis techniques. It is based on a language for specifying detailed designs, an analysis technique to create a model of a design through data-flow analysis of the language constructs, and a method to automatically generate and check properties derived from requirements to ensure a design's consistency	Can be useful outside the co-creation exercises run together with end-users. It can improve the consistency analysis only after the process of completeness analysis of requirements. It is difficult to use in agile approaches, especially where more stakeholders are involved.	[38]
A method for business modeling and elicitation of functional requirements specification. The main advantage of this method is information requirement specification adequacy as a natural way of analyzing user needs. This method is based on requirement specification, in terms of outcomes, data resources and processes. A framework that correlates the partial models with the requirement levels to	Limited to functional requirements. End-users must have background experience in the application domain. It does not cover all dimensions of consistency and there is no	[39]
increase consistency between requirements, functions, and system elements. A benefit emerging with this is the advantageous traceability of requirements.	connection with completeness analysis.	[40]
A reverse engineered meta-requirement algorithm to reverse the meta-requirements of a set of user requirements.	expertise in formalizing requirements, and comparisons with other reference systems.	[41]
A technique for generating a set of obstacle conditions guaranteed to be complete and consistent with respect to the known domain properties. The approach relies on a novel combination of model checking and learning technologies. Obstacles are iteratively learned from counterexamples and witness traces produced by model checking against a goal and converted into positive and negative examples, respectively.	The use of learning technologies at the interface by end-users with no previous experience in the application domain represents a barrier to applying this technique in the case of low-end consumer-related disruptive products.	[42]
A method for evaluation of the quality of functional requirements specifications, focusing on completeness and granularity. It relies on the definition of metrics that allow measuring certain aspects of requirement model quality.	Too difficult to be used in conjunction with non-experienced end-users, in a language that is understandable by this category of actors.	[43]
A tool that identifies user-related information items from reviews published or clicked on by a user and develops assessment models to assess the completeness of reviews based on the identified information items, and finally, ranking reviews according to the requirements for information completeness.	It works well only at a later stage of analysis, not in the early stages of product development when the goal is to identify requirements in relation to end-users that have never had previous experience in the application domain. It applies well for functional requirements on the set of requirements already defined. In some cases, non-functional requirements and categories of requirements ofter than	[44]
A technique to verify the completeness of functional requirements by enhancing the set of criteria with inputs from OTSM-TRIZ [45] concepts and tools.	functional ones are more critical for product success (e.g., in the case of disruptive products for low-end consumers, functional requirements are usually not generated in sessions with end-users, the focus being on understanding the value proposition for this category of consumers).	[46]
A goal-oriented requirements engineering tool that can model varied requirements for solutions. Different stakeholders within an enterprise may use partial or even completely non-intersecting vocabulary sets. It heavily counts on ontologies.	Less triendly in agile methodologies, where co-creation is involved from the early stages, and where conclusions in the early stages are crucial for product strategy formulation when time and budgets are fixed.	[47]
A framework rather than a method for putting together all people, processes, and the lifecycle perspective to analyze completeness of requirements.	Dedicated to well-known products in incremental innovations.	[48]
An ontology-based approach that uses Protégé software to validate completeness and consistency of requirements. It necessitates expert skills.	Very little applicable in relation to end-users with no background in the application domain and with no technical expertise.	[49]

Consistency and completeness of requirements are important aspects in the development of successful products. The ability to obtain high-quality requirement specifications from early stages of development projects generates significant benefits in terms of time and costs until a product is ready for launching on the market. Various methods have been proposed to tackle these issues, as is indicated in Table 1. However, several facets must be clarified for adopting a certain method or tool for analyzing the quality of requirements. One facet relates to the point in development at which this activity is performed. Usually, this analysis is done once the requirements collection is complete. However, there might be cases when this analysis must be done concurrently (simultaneously) with the generation of requirements (e.g., time-boxing development). Another facet refers to the category of actors involved in the process of requirements formulation. Some situations address completely new products (with no comparable reference to the market), as well as disruptive products that target market segments with no previous interaction with such products (e.g., selling a cybersecurity toolbox to a small boutique). For such cases, the involvement of end-users in the co-generation process of requirements is very challenging, not only because of a lack of previous experience but also because of the technical background and cultural background of this category of players. In such cases, before formulation of functional requirements, the focus must be on non-functional requirements and value-related requirements. Moreover, functional requirements need special attention too, in terms of usability and affordability. There are also contexts when development is strongly limited by time and money. In such cases, the ability to consolidate the database with high quality and pertinent requirements is crucial for business success. The analyzed literature in the field of requirements analysis does not provide comprehensive and end-user-centered methods for formulation of consistent and quasi-complete requirement specifications in the early stages of the project. User journey tools do not work for disruptive product innovations that address low-end consumers. Job-to-be-done techniques, one-to-one interviews, or contextual inquiry are weak for these situations. Experiments we have undertaken with representatives of end-users from this special category (i.e., non-consumers that could become low-end consumers) also indicate that the tools and the related processes proposed in the literature are too abstract for them and far away from their cultural background; thus, generating a big obstacle in involving them in the co-creation and formulation of quality requirement specifications, and further in analyzing their quality (i.e., consistency and completeness).

These conclusions motivated us to propose a new method for designing requirementspecifications for disruptive engineering solutions. The method must embed the capacity to do this job in a more natural way than those dedicated to specialists, to conduct it more flexibly (by tackling the system from any point and from several points at the same time if this is necessary), and to ensure that every iteration generates consistent and complete requirements in the predefined scope. The next section of this paper introduces this new method. It is preceded by the framework used to design the method.

## 3. Methodology

To formulate a method that better covers aspects of consistency and completeness in requirements analysis we consider several steps, which are further described in this section of the paper. The first step underlines the guiding criteria for conceptualizing the method. The second step visualizes the relationships between the guiding criteria in order to frame the scope of the investigation. The investigation is performed in the third step, by highlighting the contradictions between the guiding principles and finding the proper paths to fix these contradictions. To tackle this issue, we use the TRIZ framework [50], more specifically the so-called TRIZ contraction matrix [50]. This matrix indicates for each contraction a set of inventive principles. These inventive principles must be seen as core characteristics that the new method must embed in order to ensure that consistency and completeness will be properly covered. In other words, the first three steps deploy the performance criteria that the method must fulfill into appropriate technical characteristics. The fourth step of the methodology involves a creative process to deploy the inventive principles (the technical characteristics) into modules and interfaces (links) between modules, that all together define the new method for requirement analysis in terms of completeness and consistency.

### 3.1. Framework for Designing the New Method

The new method is going to be designed in the scope delimited by the following grounding principles: (a) to not hinder the use of any other tool in the process of requirement formulation (e.g., user journey, contextual inquiry, brainstorming, mind-mapping, job-to-be-done, interviews, customer tables, persona tables, etc.); (b) to not restrict any roadmap used for requirement formulation; (c) to be operable at any stage of requirement formulation, by moving backward and forward from that stage to cover the missing areas.

In terms of performance characteristics, the new method must be aligned to the following criteria: (a) natural use (e.g., accessible to operators with little background in the application domain); (b) capable of covering all angles that characterize consistency and completeness; (c) easy to implement in software tools for better documentation, more efficient use and facile operation in a collaborative distributed environment (both synchronous and asynchronous); and (d) flexible in terms of tackling a subject (an agile path, not a rigid one). Figure 1 illustrates the scope that frames the new method.



Figure 1. Framework for designing the new method.

To tackle the framework from Figure 1 in a structured and inventive manner, the TRIZ contradiction matrix was considered [50] (note: the TRIZ theory of inventive problem solving). This tool converts the performance vectors from Figure 1 into generic design parameters to make them suitable for application in the matrix of contradiction (a table that indicates a list of generic inventive principles for every contradiction in the design scope [50]). "Flexibility" is associated with "shape" (#12 in the TRIZ list of design parameters), "versatility" is associated with "easiness to realize the system" (#32 in the TRIZ list), "consistency" is associated with "quantity of substance" (#26 in the TRIZ list), and "completeness" is associated with "accuracy to run the system" (#29 in the TRIZ list).

"Comprehensiveness" is associated with "volume covered by the dynamic elements" (#7 in the TRIZ list), "naturalness" is associated with "convenience in use" (#33 in the TRIZ list), "diagrammatic" is associated with "clarity of the flow" (#18 in the TRIZ list), and

"agility" is associated with "adaptability of the system" (#35 in the TRIZ list). "Convenience in use" should not lead to "loss of information" (#24 in the TRIZ list). "Adaptability of the system" should not generate "effort to activate dynamic elements" (#19 in the TRIZ list). "Volume covered by the dynamic elements" should not increase the "time to execute the system" (#9 in the TRIZ list), and "clarity of flow" should not lead to higher "complexity of the tool" (#36 in the TRIZ list).

With this information and the symbolic representation of the system from Figure 1, it is possible to formulate the scope of contradictions. This thing is extremely important because solving the scope of contradiction in an inventive way of facilitating the foundation of a proper method for the purpose expressed in this research. Table 2 indicates the inventive principles associated with various conflicting problems from the design scope illustrated in Figure 1. They have been extracted from the TRIZ contradiction matrix at the intersection of the design parameters [50].

Table 2. Inventive principles for designing the new method.

Conflict	Inventive Principles
Boundary properties	
#12 versus #26	#22 Remove a harmful factor by combining it with another harmful factor #36 Use the effects that are generated during the transition phase
#12 versus #29	#32 Use "additives" to see systems or processes that are difficult to see #30 Isolate the system from its outside environment using flexible "layers"
#32 versus #26	#23 Introduce feedback #24 Mediator (use an intermediary system to do an action) #1 Increase the degree of the system's segmentation
#32 versus #29	There is no TRIZ inventive principle. This special case must be treated by considering other approaches. We have considered the SAVE method [51], which indicates consideration of principle #18 "resonance frequency".
Inner properties	
#33 versus #24	#22 Remove a harmful factor by combining it with another harmful factor #4 Replace a symmetrical system with an asymmetrical system #10 Arrange/place parts of the system in advance in such a way that they can go immediately into action when required and they do this from the most convenient position
#35 versus #19	#19 Replace a continuous action with a periodic action (an impulse) #13 Instead taking an action that is dictated by the specifications of the problem, implement an opposite action #29 Replace rigid parts of the system with reconfigurable modules that can change their "volume" or "shape"
#7 versus #9	#29 Replace rigid parts of the system with reconfigurable modules that can change their "volume" or "shape" #4 Replace a symmetrical system with an asymmetrical system #38 Do the transition from a certain level of motivation to the next higher level by enriching the system with external "motivators"
#18 versus #36	#6 Make the system perform multiple different functions; therefore, there is no need for other elements #32 Use "additives" to see systems or processes that are difficult to see #13 Instead taking an action that is dictated by the specifications of the problem, implement an opposite action

The inventive principles from Table 2 indicate a set of core technical characteristics of the method (the system) we have to design in order to embed consistency and completeness in the requirement analysis: more segmentation of the system (more modules), asymmetry (heterogenous modules, not necessarily of the same weight), capacity to combine harmful factors and generate useful effects, feedback loops, modules in resonance, system capable of more functions, capacity to reconfigure, interface layer with the outside world, etc. A close analysis of the inventive principles from Table 2 reveals properties of living systems [52]. This means that living systems have the capacity to embed consistency and completeness. This observation motivated us to dig deeper to see if the model of living systems could be a source of inspiration (by mimicry) to design the method for requirement analysis in terms of consistency and completeness. We observed that the theory of living systems has been also used for functional modelling in engineering design [53]. This finding was an additional motivator for basing the design of the new method on the living systems' model. Table 3 summarizes the subsystems of a living system, and Table 4 highlights its characteristics. They have been extracted from [52–54]. A living system comprises 19 interconnected subsystems and 9 characteristics. The architectural interconnectivity of the subsystems and the specificity of each subsystem is determined to collect and transform the matter, energy, and information from outputs that make the system adapt and survive in

the living environment. If some subsystems are missed, the system is incomplete. If some connections are missed, the system is not consistent. Therefore, analyzing requirements with a tool that replicates a living system is more effective, because such a tool intrinsically looks for consistency and completeness.

Table 3. Subsystems of a living system.

Critical Subsystem	Description	Types of Processing	Correspondent Principles from Table 2
Reproducer	The subsystem that may give rise to other similar systems The subsystem which clearly defines	Matter-energy- information	#10, #29, #6, #4
Boundary	the boundaries with other systems, which holds together the internal elements, which protects the internal elements from external stress, which excludes or allows the entry of various forms of matter–energy and information	Matter-energy- information	#30
Ingestor	The subsystem that brings matter-energy into the system from the external environment The entropy that energy the	Matter-energy	#38
Distributor	components and the system to the outside through input and output chains	Matter-energy	#10, #19, #32
Converter	The subsystem that converts inputs into useful forms for special processes in the system	Matter-energy	#22, #10, #19
Producer	The subsystem that forms stable associations for longer periods ahead of various matter and energy inputs or outputs of the converter, the results being synthesized for future useful functions (replacement of elements, repair of components, supply of energy to the supersystem, etc.)	Matter-energy	#6
Repository	The subsystem that retains in the system, for different periods of time, various forms of matter and energy	Matter-energy	#4, #10, #29
Extruder	Subsystem that transmits matter and energy outside the system in the form of products or waste	Matter-energy	#22, #19, #13, #18, #36
Motor (Engine)	and its parts in relation to the other parts or in relation to the external environment	Matter-energy	#6, #19, #24
Supporter	Subsystem that maintains adequate spatial relationships between components, allowing them to interact without pressing on each other The subsystem capable of measuring	Matter-energy	#32, #22, #10
Input transducer	the information entered into the system, to change it into another form of matter–energy compatible with the internal transmission mechanism	Information	#32, #30
Internal transducer	The subsystem which perceives the receivers from all components of the system, marks the alterations and changes them into other forms of matter-energy compatible with the internal transmission mechanism	Information	#32
Channel and net	Subsystem that directs component information through a well-defined channel or through several interconnected channels	Information	#1
Decoder	Subsystem that alters the information code transmitted from input translators to a private code, understood only within the system The subsystem that transmits to the	Information	#18
Associator	system the first lessons of the learning process to create new (perpetual) associations in order to improve the system	Information	#36, #24, #13
Memory	Subsystem that transmits lessons from the second stage of the learning process to the system and stores useful information for various periods of time	Information	#10

#### Table 3. Cont.

Critical Subsystem	Description	Types of Processing	Correspondent Principles from Table 2
Decider	Executive subsystem, which receives information (inputs) from all subsystems and transmits appropriate information (outputs) to them to control the entire system	Information	#13, #24, #23, #18
Encoder	Subsystem that alters the input information code into forms of information that can be processed by subsystems by decoding the "private" code into "public" code that can be interpreted by other systems and the external environment	Information	#36, #30, #24
Output transducer	Subsystem that posts system information markers or transforms system information into forms of matter–energy that can be transmitted through channels in the external environment	Information	#30, #40, #19, #13, #38

Table 4. Characteristics of a living system.

Characteristic	Description	Correspondent Principles from Table 2
Space-time symbiosis	A healthy living system has a balanced state in space and time; any disequilibrium provokes illness(es)	#29, #4
Matter-energy symbiosis	The right amount of matter and energy keeps the system balanced; too much or too little of matter-energy leads to problems ("starvation"/"obesity")	#6
Adequate information	Improper information, lack of information, or noise causes inadequate changes in system behavior	#32
Complete links	Missing of links leads to underperformance or blockages in the system	#1, #18
Open system	Io survive changes in the outer environment, living systems must be open and exchange information and other elements with other	#30
Supersystem (external)	A living system models and evolves to survive in the environment generated by the supersystem	#38
Feedback loop	evolutionary changes and negative loops to regulate deviations from a reference in a stable	#23
Cost and efficiency	A strong system is the one which achieves or maintains a level of performance with low consumption of resources (of any kind)	#22, #36, #24, #19, #29
Open system	io survive changes in the outer environment, living systems must be open and exchange information and other elements with other systems	#6, #13

An important remark from the analysis of information in Tables 3 and 4 is that consistency of requirements is reflected by the subsystems of a living system (see Table 3), whereas the completeness of requirements is reflected by the characteristics of a living system (see Table 4). The beauty of associating our problem with a living system stands in the fact that a living system cannot survive if it is inconsistent and incomplete in relation to the context (the external environment for which it was designed), or survives only with external support. Thus, by using the model of living systems, there are positive premises for tackling the problem of requirements formulation properly.

# 3.2. The Method for Requirements Analysis

The results from Section 3.1 indicate that a proper foundation of requirements requires 28 checkpoints. Application of the TRIZ method on the challenge "try to reduce the size of the issues" without "losing information" leads to the result that there is no guiding principle to tackle the problem in an inventive way. At its limits, TRIZ indicates that if there is no time to cover all the issues, it is important to do as much as possible. However, this compromise is not the solution. The alternative indicated by TRIZ with respect to

the above-mentioned challenge is "elastic construction". This can be translated into the following recommendation: define a complete framework for requirement definition and analysis and be elastic (flexible) in the amount of content that describes each issue.

We took the subsystems from Table 3 and organized them with respect to the characteristics from Table 4, as well as the indications from the above paragraph, to build up the new method. The goal was to generate a method that brings both end-users (non-expert users) and expert-users together to co-create (co-analyze) requirements. From this perspective, we consider that traditional approaches from system engineering are less effective because they are mainly designed by experts for experts. The canvas of the method was designed by interacting with both non-expert users and expert-users in the context of the project acknowledged at the end of this paper. The result is illustrated in Figure 2.



Figure 2. Requirement definition and analysis canvas (RDFC).

The feedback collected from the focus group involved in testing the method within the framework of the project acknowledged in this paper includes the following main aspects: (a) the canvas is comprehensive and does not allow one to omit important aspects in the requirement analysis; (b) the canvas has a logical flow; (c) the analogy with the building blocks of a living system offers to non-expert end-users the possibility to better understand the process, and to articulate their perspective on the problem under investigation. The canvas from Figure 2 (called RDFC) transposes subsystems and characteristics of a living system into a logical flow for the definition and analysis of requirement specifications; it indicates the links and the building blocks, as well as the order for tackling the process. Nevertheless, each block can be approached at any time and simultaneously with other blocks. The process is dynamic and follows an iterative (spiral) path. Schematic representation allows implementation into a software application and permits the construction of connections between the pool of requirements. More requirements can be also approached in a concurrent (simultaneous) manner. New requirements could emerge during the construction of any given requirement, too. Missing aspects can be easily identified.

# 3.3. Roadmap for Method Application

As the method proposed in this paper associates the system that is going to be designed with a living system, it is possible to break down the system in any way, and to tackle each module of the set generated by the breakdown process as a whole system. In a living system every module (note: interfaces are also modules), from the cell to the whole body, is both consistent and complete. Moreover, this association visualizes a requirement as a useful, complete, and consistent unit of the system. A living system does not embed useless units, and without external harmful actions, its body is, by nature, consistent and complete. Thus, any requirement is an organic building block of an appropriate element from the system.

Step 1: The first step of the roadmap is to declare the scope of the problem, or in other words, the part of the system you intend to focus on. We suggest basing the scope on the vision that describes the module for which you are going to define requirements. Vision is a visual description of the module, both internal and in relation to the external environment. Vision has the quality of indicating what generic units are included in the analyzed module and how they are interlinked. This leads to the creation of a starting map of units, links, constraints, and conflicts. By tackling each component from the angle of a living system, we can build up the list of requirements and even identify along the road missing units, links, constraints, or conflicts. The scope of the investigating space also reveals the stakeholders (note: from a practical point of view, the relevant stakeholders), the needs, goals, and objectives associated with the analyzed module. Additionally, applicable design regulations and standards, as well as external issues, such as higher-level requirements, other systems, budget, schedule, or technology must be associated with the scope. They can articulate drivers and constraints, which are the key elements for visualizing the context.

Step 2: The second step is to create within the scope and context the pool of scenarios, use-case, and operational concepts (all considered second-level outputs) with the involvement of representatives from all relevant stakeholders. All stages of the system's lifecycle must be considered in this respect. Both nominal and off-nominal cases must be considered. A relationship matrix between inputs (i.e., the outputs from Step 1) and second-level outputs can visualize the "pertinent" completeness. This is met when there is at least one strong connection along each row and along each column within the relationship matrix.

Step 3: The third step is to define the interfaces between the considered module and the outside "world". This is a condition imposed also by the characteristics of living systems (see Table 4).

Step 4: The fourth step is to apply the new method to construct the module's requirements. Because we operate with the concept of living systems in mind, there is no constraint from which to start this construction. In fact, it can be started simultaneously from different points. Using the method one can see at any moment the level of maturity in defining requirements with respect to each unit—the missing points. At this step, it is useful to consider all categories of requirements (i.e., functional requirements; performance (non-functional) requirements; operational requirements (all requirements dealing with how your system interacts with the outside world (external interfaces); x-ilities (including quality, safety, security); physical attributes of the system; and design and construction standards). In principle, following the set of subsystems and characteristics of a living system (see Tables 3 and 4), all these categories of requirements are implicitly captured. The beauty of this approach is that we can involve end-users in the co-creation process in a natural way because they can reflect from different angles on requirements, by associating the problem with a living system (which is quite familiar to any person, thus working on the problem without forcing their cultural mindset).

Step 5: The next step is to solve potential conflicts between the units of the analyzed module of the system. As every unit from the module fights for its growth and survival, possible conflicts might occur at the interfaces between units. This is to some degree natural but by tackling the interfaces relative to the subsystems and characteristics (Tables 3 and 4), most of the conflicts can be solved by calibrating the relationships between units, as well as between the module of the whole system. Those conflicts that cannot be solved in a natural way indicate an error in the concept of the system. Therefore, the logical path is to see where the concept is unbalanced or weak and to fix the problem. Sometimes, context is a generator of conflicts. To tackle this category of conflicts, this paper proposes to use methods for structured innovation, such as TRIZ [50], SAVE [51], or others.

Step 6: The last step is related to risk analysis. We have to accept that the process of requirements definition is framed by complexity (e.g., high entropy, nonlinearity, etc.), fuzziness (e.g., low visibility of the final solution in the early stages), and creative and intellectual effort (which depends a lot on the intellectual background of the team). Therefore, this step allows disclosing of nominal situations. These can be addressed in terms of additional requirements that facilitate the construction of a more robust system. There are more methods to tackle risks, but in this paper, we propose a simple approach to deal with risks. It comes from the AFD method [55] (note: AFD—anticipatory failure determination) and simply requires applying the principle "break out the gained accessory to the proposed solution" in order to identify weaknesses and omissions in the map. This is also aligned with the natural way of thinking because people are great at finding a weakness in a system if they are asked to find ways to destroy the system.

The next section of the paper illustrates the application of the proposed method in the case of a disruptive cybersecurity solution that addresses non-consumer (to transform them into consumers) and low-consumer markets (micro-enterprises from any economic area, from services to production). Examples of such small businesses are boutiques, accounting offices, engineering offices, consulting services, etc., as well as any kind of start-up.

## 4. Case Study

In this case study, we will not consider the whole spectrum of activities related to the roadmap presented in Section 3.3. The focus is only on illustrating the method in the fourth step of the roadmap from Section 3.3. However, to put the method in context, we have to indicate some information from step 1 of the roadmap—specifically, the scope of the problem. In this respect, we will illustrate from this step of the roadmap the value proposition, the vision, and the starting map (in the form of a mind-map) for the disruptive solution that was generated over several meeting sessions with end-users from the target group. From the vision will be extracted one need for exemplification, on which the method will be applied to illustrate the mode of application and its value. The method was used for the project acknowledged at the end of this paper. In its application, both experts from software companies and end-users were involved. This provided the opportunity to test its effectiveness through practice. Value proposition of the disruptive solution: The goal is to create a cybersecurity toolbox and related services to equip small businesses with the necessary means to defend themselves against cyber threats: (a) assess their vulnerability on cyberattacks; (b) educate employees; (c) monitor and be alert of the risks; (d) protect against cyber threats. The solution will be designed for easy and intuitive use. It will also offer close assistance from a dedicated network of experts, called "Digital Security Defenders".

Vision: To define the vision, a six-boxes canvas is proposed in this paper. It covers the following issues: (a) who (the target group); (b) what (need and opportunity); (c) what (product-service system and main functions); (d) why (value and benefits); (e) what else/even if (alternatives in the market); (f) on what (primary differentiation(s)). Figure 3 illustrates the vision in the form of a six-box canvas.

WHO (target groups)	WHAT (product-service system + main functions)
<ul> <li>SMEs and MEs (any field) - management / some employees</li> </ul>	<ul> <li>training on X levels (X=TBD)</li> </ul>
Supporters	<ul> <li>certification on X levels or Y categories</li> </ul>
Certified CS Defenders	<ul> <li>various training formats (MOOC, blended, face-to-face, online synchronous)</li> </ul>
Consultants (E-DIHs, IT Clusters, CERTs, Specialized CS Providers)	content of the toolbox
<ul> <li>Schools / Universities / Educators (students?!)</li> </ul>	<ul> <li>endpoint security solution (antimalware)</li> </ul>
Eacilitators (FEN Professional assoc. other cluster initiatives Europe Info Centers Accelerators	EDR (endpoint detection and response)
etc.)	<ul> <li>next generation firewall / IDS (hardware component)</li> </ul>
Lobbyists (EASME Dig SME Alliance etc.)	automatic software updater
	deception technology (honeypots)
WHAT (need / opportunity) *cyber-GEIGER is the name of the intended solution	encryption tools
People with modest profile in cybersecurity need close assistance to walk into the "cyber-	identity access management
labyrinth" to fix every situation, in every moment and from any place (both proactive, reactive and	• SIEM
corrective)	training platform
Need breakdown	<ul> <li>[automatic] assessment of more types of vulnerabilities</li> </ul>
<ul> <li>Intuitiveness and usability (great UX, avoid mistakes, cultural fit, etc.)</li> </ul>	<ul> <li>identify software + version that runs on each endpoint and look for known</li> </ul>
A special focus on SMEs / MEs-related IT system architectures, devices and tools	vulnerabilities in a database
<ul> <li>Community / advisory-based platform (see the LinkedIn ecosystem)</li> </ul>	scan a Web site
• Not a tool, but a Toolbox that covers all key areas (cyber-GEIGER is open to any solution provider	<ul> <li>scan a whole corporate network</li> </ul>
based on a business model, with adequate APIs in place and installation guidance)	assessment of practices, policies, procedures, rules
Toolbox includes both hardware (plug-and-play) and software solutions to meet comprehensive	<ul> <li>+ incident response policy</li> </ul>
scenarios and needs	<ul> <li>+ password policy (configuration usage storage etc.)</li> </ul>
<ul> <li>Compliance for different platforms (phone, tablet, desktops, laptops, servers, cloud interface)</li> </ul>	remote access policy
Cybersecurity testing not only on local devices / platforms, but also via remote access in the	work from home policy
company network (see distributed organizational structures in some start-ups, SMEs)	BYOD policy
Easy to understand scoring of tests	<ul> <li>propose [basic] configurations / improvements / policies (if cyber-CEIGER</li> </ul>
<ul> <li>Assess not only technical system vulnerability, but also assess good cybersecurity practices, rules.</li> </ul>	could compute a score, this means it knows which criteria are satisfied and
policies and procedures	which are not => could give some recommendations)
<ul> <li>Provides easy, accessible cybersecurity training on various skill levels, in various modes (online,</li> </ul>	<ul> <li>displays good vs. weak configurations / policies / practices</li> </ul>
face-to-face, blended)	recommends good practices for a specific SME
<ul> <li>Includes a smart virtual assistant for guidance during assessment, after assessment over the whole</li> </ul>	
lifecycle of the action plan for improvement, showing the progress, alerting for delays, indicating	·
roadmap, etc.	MUDY (select here (he)
<ul> <li>cvber-GEIGER assessment tool is smart to self-configure, based on user inputs to the particular IT</li> </ul>	WHY (value, benefits)
system (platforms, architecture, etc.)	comfort / cyber-awareness
<ul> <li>cvber-GEIGER provides also solution for guidance in case of penetrations, on priority actions.</li> </ul>	avoid/reduce loss of information
emergency plan in a dynamic way based on real-time reaction	cyber-privacy
cyber-GEIGER permanently interacts with the cybersecurity tools installed in the IT system of the	avoid/reduce loss of money
beneficiary	Increased credibility to your customers
<ul> <li>cyber-GEIGER is also a benchmarking tool (this feature increases in value at the same time with the</li> </ul>	permits benchmarking with peer companies on cyber vulnerability - this is
increasing of number of cyber-GEIGER's users (SMEs/MEs)	good for the action plan
	ON WHAT (primary differentiator(s))
WHAT ELSE / EVEN IF (alternatives in the market)	<ul> <li>It is a suite of tools for different purposes (all-in-one, covering most CS areas)</li> </ul>
<ul> <li>cybersmart (<u>https://cybersmart.co.uk/</u>) - compliance applications (Desktop, mobile, web)</li> </ul>	and not just one tool
<ul> <li>Gartner (for advisory on best tools on the market and proper reviews)</li> </ul>	Easy score for compliance purposes
<ul> <li>Nessus/Nexpose/Accunetix - Tools for internal scanning</li> </ul>	Intuitive
	Personal assistance from dedicated security engineers

Figure 3. Initial vision for the disruptive solution (called here for convenience cyber-GEIGER).

Deliberately, Figure 3 does not include the whole set of main functions generated in the exercises with end-users, both from intellectual property considerations and because it is less relevant for the purpose of this paper. The last piece of information which is required by Step 1 of the roadmap from Section 3.3 is the mind-map that indicates units, links, constraints, and conflicts. For the purpose of this paper, from the mind-map only a section is selected for illustration—the one that includes the selected need for method exemplification. Figure 4 shows this section of the mind-map. The text in Figure 4 is the one produced by a mixed team of end-users and technical specialists. This approach facilitates communication and cultural exchange, as well as guidance for end-users and compensation in areas where end-users have no idea what and how the solution should look like. It is important to perform an analysis on the text because it reflects the cultural



perspective and understanding of the end-users of the application domain (at least, from the eyes of their representatives that have participated in the discovery workshops).

Figure 4. A section from the mind-map process.

The red lines from Figure 4 indicate tensions between some areas of the generic solution. These will have to be tackled in Step 5 of the roadmap in a manner that solves conflicts in an inventive way. This issue is outside the scope of this paper.

From the map presented in Figure 4, the need to "avoid doing wrong steps" from the area "level of protection" is selected for exemplifying the application of the method proposed in Figure 2. This need is quite obvious for end-users that have no previous experience of operating tools for cybersecurity assessment. The results of applying the method to this need are illustrated in Figure 5.

The diagram from Figure 5 indicates which stakeholders have contributed to the content in each box. There are boxes where the main contributors were the end-users, others where both end-users and experts have been contributors, and boxes where experts contributed. The picture clearly shows the strong co-creation process in requirement definition and analysis. It also highlights the varying work. Thus, outputs from various boxes defined by experts are inputs for boxes where end-users and experts collaborate. If the inputs are insufficient or inappropriate from the viewpoint of the end-users, the alarm signal is activated, and things can be fixed before being rolled out in later phases. In the exercise, we involved personnel from 20 stakeholders, which are partners in the H2020 project acknowledged at the end of the paper. The background of participants from the end-user side ranges from persons with no IT background to persons with IT literacy. They belong to micro-enterprises and small businesses from various industries (i.e., advertising, accounting, B2C services, robotics) and multipliers (i.e., chambers of commerce, banks, cluster associations, professional associations). The expert-users group included architects and developers from software companies specialized in cybersecurity

(i.e., small and mid-size software companies, but also large companies, such as ATOS, Kaspersky), consulting companies, such as KPMG, and four higher education institutions from Switzerland, Germany, Holland, and Romania.



Figure 5. RDFC application on the need to "avoid doing wrong steps".

Compared with the classical use-case diagrams or boxes, the RDFC framework of the proposed method is more comprehensive, because preconditions, flows, and postconditions are better structured and visualized, thus reducing the incidence of human errors and/or omissions. Moreover, requirements are seen in a new light—wider, "alive", and more robust than they are perceived in traditional approaches.

The time allocated to fill the RDFC for the requirement illustrated in Figure 5 was about one hour. The experts involved in the experiment mentioned that the same quantity and quality of information might take double or triple the time with the traditional use-case diagrams and the risk of missing many points might be higher.

With respect to the experience captured from the exercise illustrated in Figure 5, the key boxes of the RDFC that made the difference (or made the method somehow unique) were the following: ingestor, extruder, reproducer, decider, converter, encoder, and distributor, as well as the set of nine boxes associated to the living system's characteristics (see Table 4).

### 5. Discussion

Discussions from this section are grouped into two parts: The first part provides information on the experience with the end-users; The second part comments on the method with respect to methods from other research.

The completion of the "requirement definition and analysis canvas" (RDFC) by a team of end-users and technical staff has shown that the logic flow and the topics of the building blocks of the canvas were quite intuitive for the end-users. They were capable of expressing their thoughts in a natural way. Various blocks of the canvas (e.g., input transducer, decoder, decider, etc.) that require a translation of the requirements into engineering specifications have been completed by the technical staff, but with close interaction with the end-users for ensuring that things are properly understood. The co-creation process conducted by RDFC also revealed that windows of new discoveries are opened during the completion of the canvas. The canvas's structure forces the collaboration between beneficiaries and developers in an alternating way. This minimizes the risk of perpetuating nonconformities and/or misleading points.

In this research, we did not run a systematic comparative analysis of the proposed method with respect to other methods from the state-of-the-art literature. This was not considered necessary simply because the request to design a new method was generated by a real case, enabled by a Horizon 2020 project, where various existent methods have been investigated in advance in a mixed team constructed from representatives of end-users and technical specialists. The result of discussions within the mixed team about the opportunity to select one of the existent methods for the particular case of disruptive products that address users with no previous experience in the field and with no basic background led to negative reactions from the end-users because they saw the current methods as too technical. This is actually true, because existent methods have been designed with experts in mind as target users. This was in fact the trigger for investigating a new(more friendly) way of carrying out collaborative work in agile methodologies to design new products.

The quality of the proposed method for adequate coverage of completeness and consistency in the case of requirement analysis is systematized in Table 5. Issues highlighted in Table 5 are the questions that describe consistency and completeness.

Table 5 shows in the argumentation column how different blocks of the canvas work for fulfilling the need for consistent and complete definition and analysis of a given requirement. The case study highlights in Figure 5 how various stakeholders contribute to the creation of the content in the canvas. There are blocks where end-users are involved, blocks where only experts are involved, and blocks where both end-users and experts must collaborate. It is important to highlight that the outputs of a block are inputs in another block or other blocks. This relates to the internal "client–supplier" concept within the canvas, thus supporting the quality of the process, because if some "supplier" blocks do not deliver adequate outputs this is signaled by other "client" blocks.

Issue	Arguments
Is all pertinent information included?	Figures 2 and 5 show the canvas (the general form and the application for defining a requirement). The 19 blocks and 9 characteristics of the canvas cover all aspects that refer to a complete system, its interfaces with the external environment, and its internal links. In the case study from Figure 5 it is also highlighted the fact that all stakeholders are involved. By covering all blocks of the canvas, the pertinent information shall be included in the description of
Which is the significance of the judged requirement?	a requirement. The blocks called "Decoder" and "Converter" are responsible for this job (see Figure 2)
Which is the measuring unit and on which scale is it measured?	The block called "Ingestor" is responsible for this issue (see Figure 2).
How will the measurement process be performed to receive trustworthy information?	There are several blocks in the canvas responsible for measurement. They are called "Transducers". It is important to highlight that we have several types of transducers in the canvas, both at the interface with the outer environment and internal, as well as encoders and decoders. The "Decider" block is responsible for handling trustworthy information (see Figure 2)
What is the goal in terms of requirement fulfilment?	The block called "Boundary" covers this issue (see Figure 2).
What must be the minimum level of achievement?	The block called "Output transducer" and the set of 9 blocks that refers to characteristics (bottom side of the canvas) covers this question (see Figure 2). It also relates with the "Channel" block and "Motor" block.
How much is going to be achieved in a well-defined time (nominal value and tolerance)?	The block "Decider", in relation with "Producer", "Supporter", and "Reproducer" tackle this issue (see Figure 2).
How is the information about evolution to be recorded?	The "Memory" and "Repository" blocks are responsible for this job (see Figure 2).
What is the level of achievement in other comparable systems and why?	The "Distributor", "Associator", and "Reproducer" blocks in association with "Ingestor" and "Extruder" blocks are designed for this purpose (see Figure 2).

Table 5. Analysis of the proposed method.

#### 6. Conclusions

Requirement design and analysis in the development process of brand new, disruptive products is a very challenging job. Engineers have to extract as much pertinent information as possible from end-users and customers (buyers) in the early stage(s) of the project if the time and budget are rigid. This happens in many cases of research & development projects that are run by start-ups with public funds or public–private funds, or even with private funds. Under these circumstances, the agile methodology is subject to the "time and cost boxing" constraint. This is interpreted in the key dictated by the concurrent engineering concept, which calls for high quality and comprehensive planning at the start of the project, when actually very little information is available. This is obvious because the product is "invisible" at that stage. Thus, the product must be associated with a "substitute" (a virtual twin) to do the job.

The paradigm of living systems was found in this research as a plausible framework to build up the substitute, because it is known that a living system (simple or complex) incorporates both consistency and completeness at each scale. This offers a natural way to construct a substitute product (built from requirements) starting from simple forms to complex forms in an organic way. Moreover, the paradigm of the living system is easier understood by non-technical persons because everyone can associate this with something already known and experienced. Therefore, the consistent involvement of end-users in the early stages of the project is much more feasible than when using abstract, techniciantailored tools. The method introduced in this paper is composed of a canvas that replicates the subsystems and logical flow of a living system, and a six-step roadmap to integrate the canvas during the process of requirement analysis. The research question is properly addressed by this method; a detailed proof is shown in Table 5. By its structure and flow, the method tackles the research question in a natural way, thus allowing the involvement of non-IT experts (i.e., end-users) in the co-creation process of requirement definition and analysis.

The proposed method might be subject to criticism, too. For some people, it might look too "complete", with many boxes. The truth is that nobody can skip this job; either choose to plan more at the beginning, with the premise of reducing the risk of meeting nonconformities in the later stages of the project, or leave this job for later phases, with the risk of running out of time and budget. Additionally, the method might undergo improvements during its application in more and diverse projects. Until this moment, we succeeded in proving its effectiveness in one project, which is acknowledged at the end of the paper. This project involves a consortium of mature software companies, universities, and beneficiaries (end-users). Both researchers, experts, practitioners, and end-users were involved, and the final form of the method (as it is presented in this paper) includes the feedback provided by these stakeholders during its application in the project.

We see no barrier to applying this method for any category of products, from pure mechanical to electro-mechanical, mechatronic and software, as well as for services, or combinations of products and services (PSS)—although the single opportunity we've had up until now to investigate its capability was in a software development project, this method has much more potential for application if it is implemented in a software tool. This can open new windows of opportunity to refine the method in the sense of handling the content, making analysis of consistency and completeness automatic, and including modern algorithms of natural language processing.

**Author Contributions:** This paper had equal contributions of all authors. Conceptualization, S.B.; methodology, S.B. and E.B.; software, S.B.; validation, E.B.; formal analysis, E.B. and S.B.; investigation, E.B.; resources, S.B.; data curation, E.B.; writing—original draft preparation, S.B.; writing—review and editing, E.B.; visualization, E.B.; supervision, S.B.; project administration, E.B.; funding acquisition, S.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by European Commission, grant number 883588 GEIGER project, under the H2020 program, which is acknowledged with gratitude.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

## References

- Maguire, M.; Bevan, N. User requirements analysis: A review of supporting methods. In *Usability: Gaining a Competitive Edge*; Hammond, J., Gross, T., Wesson, J., Eds.; IFIP WCC TC13 2002; IFIP—The International Federation for Information Processing; Springer: Boston, MA, USA, 2002; Volume 99, pp. 133–148. [CrossRef]
- 2. Maciaszek, L.A. Requirements Analysis and System Design, 3rd ed.; Addison-Wesley: Harlow, England, 2007; pp. 26–39.
- 3. Hay, D.C. *Requirements Analysis: From Business Views to Architecture;* Pearson Education: Upper Saddle River, NJ, USA, 2003; pp. 1–9.
- Bahill, A.T.; Madni, A.M. Discovering system requirements. In *Tradeoff Decisions in System Design*; Bahill, A.T., Madni, A.M., Eds.; Springer: Cham, Switzerland, 2017; pp. 373–457. [CrossRef]
- Jarke, M.; Lyytinen, K. Complexity of system evolution: Requirements engineering perspective. ACM Trans. Manag. Inf. Syst. 2015, 5, 1–7. [CrossRef]
- 6. Vezzoli, C.A. *Design for Environmental Sustainability: Life Cycle Design of Products,* 2nd ed.; Springer: Cham, Switzerland, 2018; pp. 209–297.
- Zhang, X.; Zhang, L.; Fung, K.Y.; Bakshi, B.R.; Ng, K.M. Sustainable product design: A life-cycle approach. *Chem. Eng. Sci.* 2020, 217, 115508. [CrossRef]
- 8. Falkenreck, C. Design thinking—Interactively co-creating innovative products that fit the market. In *Creativity and Marketing: The Fuel for Success;* Pantano, E., Ed.; Emerald Publishing Limited: Bingley, England, 2021; pp. 69–84. [CrossRef]
- Potra, S.; Pugna, A.; Negrea, R.; Izvercian, M. Customer perspective of value for innovative products and services. *Procedia Soc. Behav. Sci.* 2018, 238, 207–213. [CrossRef]
- 10. Lin, C.C.; Luh, D.B. A vision-oriented approach for innovative product design. Adv. Eng. Inform. 2009, 23, 191–200. [CrossRef]
- 11. Altameem, E.A. Impact of agile methodology on software development. Comput. Inf. Sci. 2015, 8, 9–14. [CrossRef]
- Nazir, A.K.; Zafar, I.; Abbas, M. The impact of agile methodology (DSDM) on software project management. Circulation in Computer Science (International Conference on Engineering, Computing & Information Technology (ICECIT)), Kuala Lumpur, Malaysia, 20–22 October 2017; pp. 1–6.

- 13. Kendall, K.; Kong, S.; Kendall, J. The impact of agile methodologies on the quality of information systems: Factors shaping strategic adoption of agile practices. *Int. J. Strateg. Decis. Sci.* **2010**, *1*, 16. [CrossRef]
- 14. Srivastava, A.; Bhardwaj, S.; Saraswat, S. SCRUM model for agile methodology. In Proceedings of the 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, India, 5–6 May 2017; pp. 864–869.
- 15. Alshamrani, A.; Bahattab, A. A comparison between three SDLC models: Waterfall model, spiral model, and incremental/iterative model. *Int. J. Comput. Sci.* 2015, *12*, 106–111.
- 16. Brad, S.; Murar, M.; Brad, E. Methodology for lean design of disruptive innovations. Procedia CIRP 2016, 50, 153–159. [CrossRef]
- Godoy, C.P.; Cruz, A.F.; Silva, E.P. Blueprint model: A new approach to SCRUM agile methodology. In Proceedings of the 2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE), Montreal, QC, Canada, 25–26 May 2019; pp. 95–99.
- 18. Yamazaki, K. Proposal for spiral design approach. In Proceedings of the Second International Symposium on Environmentally Conscious Design and Inverse Manufacturing, Tokyo, Japan, 11–15 December 2001; pp. 137–142. [CrossRef]
- Dombrowski, U.; Schmidt, S. Integration of design for X approaches in the concept of lean design to enable a holistic product design. In Proceedings of the 2013 IEEE International Conference on Industrial Engineering and Engineering Management, Bangkok, Thailand, 10–13 December 2013; pp. 1515–1519. [CrossRef]
- 20. Nikolaev, M.Y.; Fortin, C. A literature review of design decision making in disruptive technological innovations of new products. In Proceedings of the ASME 2020 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (Volume 6: 25th Design for Manufacturing and the Life Cycle Conference (DFMLC)), Online, 17–19 August 2020. [CrossRef]
- Kumar, A.; Lokku, D.S.; Natarajan, S.; Nori, K.V. Architecting disruptive digital product-service systems. *INCOSE Int. Symp.* 2018, 28, 1280–1295. [CrossRef]
- 22. Lindskog, E.; Berglund, J.; Vallhagen, J.; Johansson, B. Visualization support for virtual redesign of manufacturing systems. *Procedia CIRP* **2013**, *7*, 419–424. [CrossRef]
- 23. Sadokierski, Z. Developing critical documentation practices for design researchers. Des. Stud. 2020, 69, 100940. [CrossRef]
- 24. Lucassen, G.; van der Keuken, M.; Dalpiaz, F.; Brinkkemper, S.; Sloof, G.W.; Schlingmann, J. Job-to-be-done oriented requirements engineering: A method for defining job stories. In *Requirements Engineering: Foundation for Software Quality. REFSQ 2018. Lecture Notes in Computer Science;* Kamsties, E., Horkoff, J., Dalpiaz, F., Eds.; Springer: Cham, Switzerland, 2018; Volume 10753, pp. 227–243. [CrossRef]
- 25. Bullock, A. How to conduct one-to-one qualitative interviews for research. Educ. Prim. Care 2016, 27, 330–332. [CrossRef]
- Nielsen, L.; Hansen, K.S.; Stage, J.; Billestrup, J. A template for design personas: Analysis of 47 persona descriptions for Danish industries and organizations. *Int. J. Sociotechnol. Knowl. Dev.* 2015, 7, 17. [CrossRef]
- 27. Endmann, A.; Kessner, D. User journey mapping—A method in user experience design. *J. Interact. Media* **2016**, *15*, 408–414. [CrossRef]
- Verma, K.; Kass, A. Requirements analysis tool: A tool for automatically analyzing software requirements documents. In *ISWC* 2008; Sheth, A., Ed.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 751–763.
- Van Eck, N.J.; Waltman, L. Visualizing bibliometric networks. In *Measuring Scholarly Impact: Methods and Practice*; Ding, Y., Rousseau, R., Wolfram, D., Eds.; Springer: Cham, Switzerland, 2014; pp. 285–320.
- Futrell, R.T.; Shafer, D.F.; Shafer, L.I. Quality Software Project Management; Prentice Hall: Upper Saddle River, NJ, USA, 2005; pp. 73–99.
- 31. Arora, C.; Sabetzadeh, M.; Briand, L.C. An empirical study on the potential usefulness of domain models for completeness checking of requirements. *Empir. Softw. Eng.* 2019, 24, 2509–2539. [CrossRef]
- Bloem, R.; Cimatti, A.; Greimel, K.; Hofferek, G.; Konighofer, R.; Roveri, M.; Schuppan, V.; Seeber, R. RASTY—A new requirements analysis tool with synthesis. In *CAV 2010*; Touili, T., Cook, B., Jackson, P., Eds.; Springer: Berlin/Heidelberg, Germany, 2010; pp. 425–429.
- Shimomura, Y.; Nemoto, Y.; Ishii, T.; Nakamura, T. A method for identifying customer orientations and requirements for product-service systems. *Int. J. Prod. Res.* 2018, 56, 2585–2595. [CrossRef]
- 34. Beringer, S.; Wehrheim, H. Consistency analysis of AUTOSAR timing requirements. In Proceedings of the 15th International Conference on Software Technologies; Vansinderen, M., Fill, H.G., Maciaszek, L., Eds.; ICSOFT: Setubal, Portugal, 2020; pp. 15–26.
- 35. Heimdahl, M.P.; Leveson, N.G. Completeness and consistency in hierarchical state-based requirements. *IEEE Trans. Softw. Eng.* **1996**, *22*, 363–377. [CrossRef]
- 36. Filipovikj, P.; Rodriguez-Navas, G.; Nyberg, M.; Seceleanu, C. Automated SMT-based consistency checking of industrial critical requirements. *Appl. Comput. Rev.* 2017, 17, 15–28. [CrossRef]
- Li, X.S.; Liu, Z.M.; He, J.F. Consistency checking of UML requirements. In Proceedings of the 10th IEEE International Conference on Engineering of Complex Computer Systems-ICECCS, Shanghai, China, 16–20 June 2005; IEEE Computer Society: Los Alamitos, CA, USA, 2005; pp. 411–420.
- 38. Chechik, M.; Gannon, J. Automatic analysis of consistency between requirements and designs. *IEEE Trans. Softw. Eng.* 2001, 27, 651–672. [CrossRef]

- Butkiene, R.; Butleris, R.; Danikauskas, T. The approach to consistency checking of functional requirements specification. In Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics, Orlando, FL, USA, 14–18 June 2002; Callaos, N., Zhao, R., Sanchez, B., Lesso, W., Eds.; International Institute of Informatics and Systemics: Winter Garden, FL, USA, 2002; Volume 18, pp. 67–72.
- Lankeit, C.; Just, V.; Traechtler, A. Consistency analysis for requirements, functions, and system elements requirements for the entire development process. In Proceedings of the 2016 Annual IEEE Systems Conference, Orlando, FL, USA, 18–21 April 2016; pp. 499–504.
- 41. Ong, M.I.; Ameedeen, M.A.; Kamarudin, I.E. Meta-requirement method towards analyzing completeness of requirements specification. In Proceedings of the Future Technologies Conference (FTC), Vancouver, BC, Canada, 13–14 November 2018; Arai, K., Bhatia, R., Kapoor, S., Eds.; Springer: Cham, Switzerland, 2018; Volume 881, pp. 444–454.
- Alrajeh, D.; Kramer, J.; van Lamsweerde, A.; Russo, A.; Uchitel, S. Generating obstacle conditions for requirements completeness. In Proceedings of the 34th International Conference on Software Engineering, Zurich, Switzerland, 2–9 June 2012; Glinz, M., Murphy, G., Pezze, M., Eds.; IEEE: New York, NY, USA, 2012; pp. 705–715.
- Espana, S.; Condori-Fernandez, N.; Gonzalez, A.; Pastor, O. Evaluating the completeness and granularity of functional requirements specifications: A controlled experiment. In Proceedings of the 17th IEEE International Requirements Engineering Conference, Atlanta, GA, USA, 30 August–4 September 2009; pp. 161–168.
- Li, C.; Zhang, H.; Xiang, J.; Li, F. The completeness assessment on product reviews based on user-concerned information requirements. In Proceedings of the Third International Conference on Computer Science and Application Engineering, Sanya, China, 22–24 October 2019; Emrouznejad, A., Ed.; Assoc Computing Machinery: New York, NY, USA, 2019; pp. 1–5.
- 45. Becattini, N.; Cascini, G.; Rotini, F. OTSM-TRIZ network of problems for evaluating the design skills of engineering students. *Procedia Eng.* **2015**, *131*, 689–700. [CrossRef]
- Becattini, N.; Cascini, G. General-purpose requirements checklist for improving the completeness of a design specification. In Proceedings of the 13th International Design Conference, Dubrovnik, Croatia, 19–22 May 2014; Marjanovic, D., Storga, M., Pavkovic, N., Bojcetic, N., Eds.; Design Soc: Glasgow, Scotland, 2014; pp. 111–120.
- Deb, N.; Chaki, N.; Ghose, A. Using i\* model towards ontology integration and completeness checking in enterprise systems requirement hierarchy. In Proceedings of the IEEE International Workshop on Model-Driven Requirements Engineering, Ottawa, ON, Canada, 24 August 2015; Mussbacher, G., Araujo, J., Moreira, A., Sanchez, P., Eds.; IEEE: New York, NY, USA, 2015; pp. 11–20.
- Zenun, M.M.; Loureiro, G. A framework for completeness in requirements engineering: An application in aircraft maintenance scenario. In Proceedings of the 20th ISPE International Conference on Concurrent Engineering, Melbourne, Australia, 2–6 September 2013; Bil, C., Mo, J., Stjepandic, J., Eds.; IOS PRESS: Amsterdam, The Netherlands, 2013; pp. 568–577.
- 49. Avdeenko, T.; Pustovalova, N. The ontology-based approach to support the completeness and consistency of the requirements specification. In Proceedings of the International Siberian Conference on Control and Communications, Omsk, Russia, 21–23 May 2015.
- 50. Altshuller, G. *The Innovation Algorithm. TRIZ*; Technical Innovation Center: Worcester, MA, USA, 2000.
- 51. Brad, S. Structured activation of vertex entropy (SAVE): Another way around creative problem solving for non-technical applications. *Innov. J. Eur. TRIZ Assoc.* **2017**, *3*, 76–81.
- 52. Miller, J. Living systems. Q. Rev. Biol. 1973, 48, 63–91. [CrossRef] [PubMed]
- 53. Cowan, F.S.; Allen, J.K.; Mistree, F. Functional modelling in engineering design: A perspectival approach featuring living system theory. *Syst. Res. Behav. Sci.* 2006, 23, 365–381. [CrossRef]
- 54. Latsu-Dake, E.; Ntuen, C.A. A conceptual model for designing adaptive human-computer interfaces using the living systems theory. *Syst. Res. Behav. Sci.* 2009, *26*, 15–27. [CrossRef]
- 55. Chen, J.L.; Hung, C. Eco-innovation by anticipatory failure determination (AFD) method. In Proceedings of the Design Society: International Conference on Engineering Design, Delft, The Netherlands, 5–8 August 2019; Volume 1, pp. 3271–3280.