

A Metamodel for Agile Requirements Engineering

Eva-Maria Schön¹, Jorge Sedeño¹, Manuel Mejías¹, Jörg Thomaschewski², María José Escalona¹

¹Department of Computer Languages and Systems, University of Seville, Seville, Spain

²Faculty of Technology, University of Applied Sciences Emden/Leer, Emden, Germany

Email: eva.schoen@iwt2.org

How to cite this paper: Schön, E.-M., Sedeño, J., Mejías, M., Thomaschewski, J. and Escalona, M.J. (2019) A Metamodel for Agile Requirements Engineering. *Journal of Computer and Communications*, 7, 1-22. <https://doi.org/10.4236/jcc.2019.72001>

Received: October 24, 2018

Accepted: January 26, 2019

Published: January 29, 2019

Copyright © 2019 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Value delivery is becoming an important asset for an organization due to increasing competition in industry. Therefore, companies apply Agile Software Development (ASD) to be more competitive and reduce time to market. Using ASD for the development of systems implies that established approaches of Requirements Engineering (RE) undergo some changes in order to be more flexible to changing requirements. To this end, the field of agile RE is emergent and different process models for agile RE have arisen. The aim of this paper is to build an abstract layer about the variety of existing process models by means of a metamodel for agile RE. It has been created in several iterations and relies on the evaluation of related process models. Furthermore, we have derived process models for agile RE in industry by presenting instances of the metamodel in two different cases: one is based on Scrum whereas the other is based on Kanban. This paper contributes to the software development body of knowledge by delivering a metamodel for agile RE that supports researchers and practitioners modeling and improving their own process models. We can conclude that the agile RE metamodel is highly relevant for the industry as well as for the research community, since we have derived it following empirical research in the field of ASD.

Keywords

Agile Software Development, Requirements Engineering, Human-Centered Design, Metamodel

1. Introduction

Agile Software Development (ASD) is commonly used in environments where people have to solve complex adaptive problems. Agile methodologies (e.g.

Scrum [1], Kanban [2] or Extreme Programming (XP) [3]) are often combined with Human-Centered Design (HCD) activities [4] when applying them in industry. One objective of this integration of methodologies is to evolve to a value-driven organization where systems can be developed to fulfil user needs by offering a positive User Experience (UX) [5].

Focusing on value delivery becomes an emergent aspect in system development due to the increasing competition in industry. Therefore, established Requirements Engineering (RE) approaches need to be adapted to the requirements of ASD. Analyzing existing literature reviews related to agile RE [5] [6] [7] [8], we can observe that a couple of authors propose different process models for agile RE. Such process models have in common the agile values on which they rely [9]:

- Individuals and interactions over processes and tools;
- Working software over comprehensive documentation;
- Customer collaboration over contract negotiation;
- Responding to change over following a plan.

Nevertheless, these process models for agile RE differ in how the way of working is described. For instance, which methods or artifacts are recommended to use for discovering, refining, prioritizing, reviewing or documenting requirements in an agile environment. We can build an abstract view on this field by means of a metamodel so as to gain a better understanding of agile RE and reduce the complexity of existing process models. Moreover, the analysis of related work shows that the description of agile RE management in the literature varies a lot from each other and that there is no common language. The metamodel for agile RE helps us to describe key concepts of agile RE management. This gives us a common definition for this research area and supports us in the development of a unified language. As the environment where agile is used is complex, due to the number of diverse people working together in different roles to create one system, we need an extremely adaptive metamodel for describing agile RE on an abstract level.

This paper presents a metamodel for agile RE (agile RE metamodel), aiming to build an abstraction layer about the variety of existing process models in the field. The agile RE metamodel allows building a common understanding of the complex research field of agile RE. In addition, we provide two instantiations to domain specific models for agile RE by means of two cases run in industry: i) case a discusses on a Scrum-based process model for developing several systems in the E-government area; and ii) case b presents a Kanban-based process model used to relaunch an internet-based newspaper portal.

The paper is structured as follows: Section 2 summarizes the related work and points out the research gap. Then, Section 3 describes the applied method for creating and instantiating the metamodel. Section 4 presents the metamodel for agile RE. Later, Section 5 discusses on the instantiation into process models. Section 6 analyzes the results and limitations of this research and finally, Section 7 states conclusions and provides an outlook on future research.

2. Related Work

We conducted a systematic literature review [5] in order to analyze the state-of-the-art of agile RE with strong focus on stakeholder and user involvement. In particular, we investigated what approaches exist to involve stakeholder in the process, which methodologies are commonly used to present the user perspective and how requirements management is carried out. In sum, 27 papers were included in the study [5] and analyzed according to our predefined research protocol.

The results of the state-of-the-art analysis revealed that the research field of agile RE is very close to current work practices in companies, since most of the included studies report results from case studies (19/27 = 70%). In addition, we observed that agile RE is a complex research field with a lot of different cross-functional influences, for instance, from the fields of HCD, ASD or RE.

However, we also identified some gaps in present literature. We learned that building a shared understanding concerning the user perspective is not very well established in agile environments. Thus, we were not able to find a common process model for stakeholder and user involvement. Nevertheless, organizations deal with those shortcomings by integrating additional methodologies like HCD, Design Thinking, Contextual Inquiry or Participatory Design.

Furthermore, we found studies presenting process models for agile RE [10] [11] [12] [13] [14]. All these process models share the continuous management of requirements by involving stakeholder and user. The most related ones are highlighted below.

Memmel *et al.* [10] develop a XP-based process model called Cross-discipline User Interface and Software Engineering (CRUISER). The process starts with an Initial Requirements Up-front Phase (IRUP) whose results are agile models that describe user needs by means of agile techniques (e.g. essential use cases, scenarios and prototypes). The gathered information is elaborated and processed during the different phases of CRUISER.

Kautz [11] carried out a case study with the aim to investigate user and customer involvement in ASD. He does not claim to propose a process model but the results of the study present an implicitly applied process model. Kautz integrates Participatory Design activities in XP. The agile team can detect problems concerning misunderstanding of requirements early before they can grow into larger problems by the application of an onsite customer as well as iteratively reviews with users and customers.

Maguire [12] extends the HCD [4] framework for ASD. The HCD process consists of the following steps: plan the HCD process, understand and specify the context of use, describe user requirements, produce design solutions to meet such requirements and evaluate designs against them. He suggests a set of agile techniques that can be used in each step. Besides, he proposes a couple of artifacts that are generated while using agile techniques.

Rivero *et al.* [13] propose a process model known as Mockup-Driven Del-

opment (MockupDD). Their approach supports Model-Driven Web Engineering (MDWE) and is integrated into Scrum. In the beginning of MockupDD, a quick requirements gathering stage is performed resulting in a set of user stories. Based on this, costumers and users create mockups to represent these user stories graphically. These mockups lay the foundation for the following modeling process.

Olsson *et al.* [14] design a process model based on a conceptual model named Qualitative/Quantitative Customer-Driven Development. It stresses the need for combining qualitative customer feedback in early stages of development with quantitative observations in later stages. Olsson *et al.* treat requirements as hypotheses that are validated with customers before development. Hypotheses are derived from business strategies, innovation initiatives, customer feedback and on-going validation cycles.

Analyzing the commonalities of the proposed approaches, we can conclude that all these process models have in common that they utilize different types of artifacts, meetings, methods and roles (referred to as agile techniques) for the management of requirements in an agile environment. In this context, the related work lacks in: 1) providing guidelines that help to choose an appropriate set of agile techniques for solving a specific agile RE problem [15]; and 2) providing generic concepts on an abstraction layer. These generic concepts are important in today's business world since companies utilize different types of process models for diverse teams.

This leads to increasing complexity within an organization (e.g. scaled organization or other teams working with sequential approaches, like waterfall model). This paper proposes a metamodel for agile RE to handle this complexity (see Section 4). It is evaluated by an instantiation into concrete process models (see Section 5). To the best of our knowledge this is the first metamodel for agile RE.

3. Research Method

The objectives of this paper are contributing a metamodel for agile RE, which allows building a common understanding of the complex research field of agile RE; and showing how the metamodel can be used in industry by means of tailoring domain specific models.

In the following subsections, we will explain how we developed the agile RE metamodel iteratively. Subsequently, we will demonstrate how the metamodel can be applied to industry by means of instantiations.

3.1. Developing the Agile RE Metamodel

Figure 1 displays the phases that were applied to develop the agile RE metamodel. In phase 1, we analyzed existing process models for agile RE in order to describe agile RE on an abstract level. Based on the related work and our experience working with requirements in agile environments, we created the initial version of the agile RE metamodel (phase 2), which was presented to the re-

search community and then discussed [16]. This metamodel has been optimized in several iterations by means of discussions with other researchers and practitioners in the field of agile RE as well as further empirical studies [15].

The discussions about the metamodel triggered first improvements. To this end, we started to elaborate the agile RE metamodel iteratively based on our learnings. This approach is similar to the concepts of ASD, where products are created in several iterations. **Table 1** shows an overview of the conceptual improvements of the agile RE metamodel in accordance with the events and activities that triggered these improvements. The current version of the metamodel is presented in **Figure 3**.

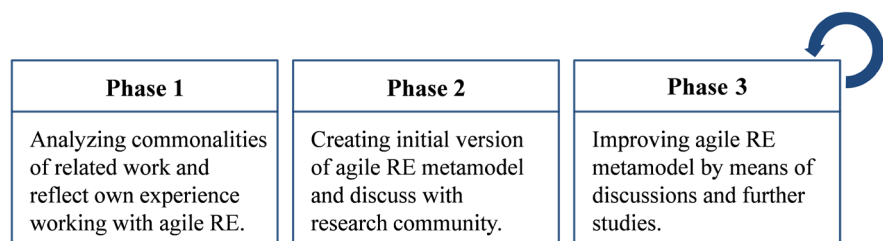


Figure 1. Phases to create the agile RE metamodel.

Table 1. Overview of conceptual improvements of the agile RE metamodel.

Version	Trigger	Improvements
v1.1	Learning from doctoral consortium [16]	Addition of relations between metaclasses and extension of attributes in several metaclasses according to ISO 9241-210: 2010 [4]
v1.2	Discussion with Dr. Jorge Sedeño López	Extension of attributes and changing names
v1.3	Discussion with Jutta Doetkotte in relation to her bachelor thesis	Extension of attributes and changing names, types and relations among metaclasses
v1.4	Learning from the study “Key Challenges in Agile RE” [15]	Extension of relations, deletion of attributes, change types of attributes, reorder attributes within metaclasses
v1.5	Learning from the study “Identifying Agile RE Patterns” [17]	Change attributes and extension of attributes
v1.6	Discussion with Dr. María José Escalona, Dr. Jörg Thomaschewski and Dr. Manuel Mejías	Deletion of attributes, change of types of attributes according to UML notation and addition of new metaclasses
v1.7	Discussion with Dr. Nora Koch, Dr. María José Escalona and Dr. Jörg Thomaschewski	Extension of relations, deletion of attributes and addition of metaclass Requirement
v1.8	Discussion with Dr. Nora Koch, Dr. María José Escalona and Dr. Jörg Thomaschewski	Definition of metaclass requirement as external class, update of relations and multiplicities, change of the metaclasses ordering and deletion of redundant attributes

3.2. Tailoring the Agile RE Metamodel to Domain Specific Models

The agile RE metamodel needs to be instantiated in terms of conditions of the specific organizational environment, in order to be used in industry. This step is related to the approach by Rolland [18], who presents a technique for process modeling by means of building abstraction levels (process meta-level, process model and development runs). Following this approach enables us to define how to apply the agile RE metamodel into practice. Therefore, we have created instances of the metamodel (L2) to derive domain specific models (L1) that could be utilized in industry. It is worth highlighting that we have slightly adapted the approach by Rolland to become more systematic. Hence, we recommend using a profile for building instances of the agile RE metamodel [19].

In this paper, we will present two practical cases for the instantiation of the agile RE metamodel. On the one hand, we will exemplify a process model for a Scrum-based process regarding the development of web applications in E-government areas. The result will be a process for discovering services in the early stages of agile development (Section 5.1). On the other hand, we will derive an instantiation of the agile RE process metamodel into a Kanban-based process model for developing an internet-based newspaper portal (Section 5.2). We have decided to use these two cases, because they show the adaptability of the agile RE metamodel in terms of using different agile methodologies.

4. Agile RE Metamodel

The agile RE metamodel (see **Figure 3**) visualizes the big picture of agile RE without providing a concrete domain specific model. We can use it to derive a domain specific model tailored according to the environment where the system is developed. Moreover, the agile RE metamodel represents a set of generic concepts without describing a domain specific model itself. Therefore, it builds a framework for a process. Hence, the domain specific model is an instantiation of the metamodel (see **Figure 2**).

The agile RE metamodel looks at the cross-functional area of agile RE that consists of HCD [4] (pink, metaclasses Context of Use and User), ASD (green, metaclasses Stakeholder, Agile Team, Methodology and Impact), Domain Driven

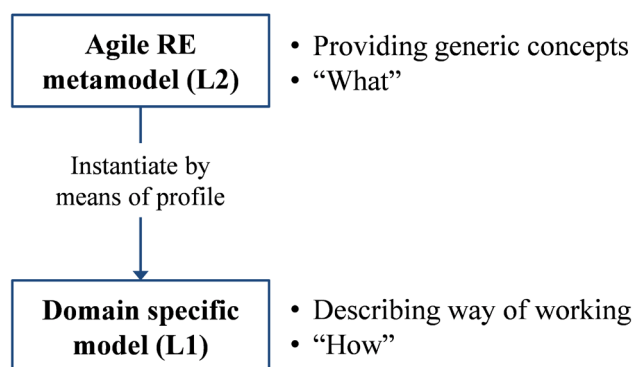


Figure 2. Modeling agile RE by instantiating the agile RE metamodel.

Design [20] (DDD, orange, metaclasses System and Domain) and RE [21] (grey and metaclass Agile RE Activity). The blue metaclasses (Organizational Environment, Agile RE Problem, Agile RE Pattern and Requirements) are part of our contribution to the body of knowledge of agile RE. The initial version of the agile RE metamodel is presented in [16], whereas **Figure 3** represents the optimized version.

4.1. Description of the Agile RE Metamodel

The metamodel contains the metaclass User with the following attributes: name, personal information, behavior, usage behavior, motivation and pain points. The user is in a context of use during the usage of the system. Thus, s/he is working with specific equipment in a physical environment while carrying out tasks. Moreover, s/he interacts in a social environment. In ASD, and for instance in XP, the user can be part of the agile team, which includes name, members, size and role. There can be more than just one agile team in a scaled environment.

The continuous involvement of stakeholders is very important to ASD. The metaclass Stakeholder includes the attributes name, group, need and priority, where need is related to human needs.

Another important aspect in agile RE is the applied agile methodology (e.g. Scrum, Kanban or XP). Each methodology comes with its own requirements and

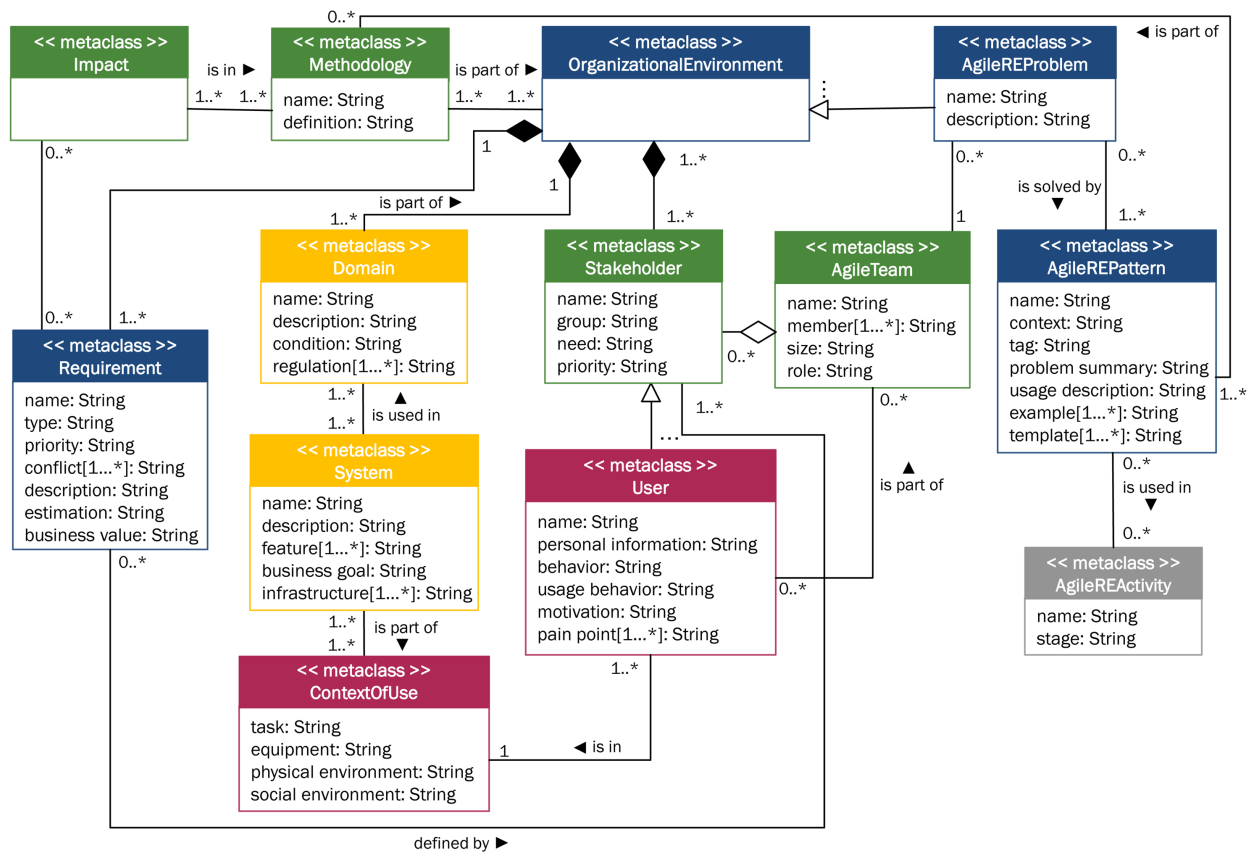


Figure 3. Agile RE metamodel.

has an impact on how RE is carried out. For instance, there is a difference between iterative approaches like Scrum or flow-driven approaches like Kanban. Therefore, we add the metaclass Impact that is filled at runtime. This metamodel defines the system to be developed with the following attributes: name, description, feature, business goal and infrastructure. The requirements for each system differ due to the diverse domains in which the system is used. The metaclass Domain becomes more important in terms of the era of Industry 4.0. As an example, it can be highlighted that there are a lot of different Internet of Things (IoT) platforms on the market that need to be customized for a specific domain (e.g. automotive, utilities or wind energy). The domain is described within the scope of the metamodel with name, description, condition and regulation.

All these metaclasses are influencing each other and the outcome is described as metaclass Organizational Environment (completed at runtime). On the one hand, the organizational environment is changing because of the different people involved in the process (user, stakeholder or agile team). On the other hand, it is changing because of the domain or the agile methodology used.

As visualized by the agile RE metamodel, agile RE problems (metaclass Agile RE Problem) occur in an agile organizational environment (metaclass Organizational Environment). Agile RE problems are solved by agile RE patterns (metaclass Agile RE Pattern). Therefore, the metaclass Agile RE Problem is generalized by the metaclass Organizational Environment. Agile RE problems occur during system development in an agile context and they are defined by means of name and description. Examples of agile RE problems can be “continuous management of requirements” (P4) or “not to lose sight of the big picture during the implementation of complex requirements” (P3) (see **Table 3**).

Besides, an agile RE problem is solved by one or more agile RE patterns. The agile RE problem “continuous management of requirements” can be solved by the agile RE patterns “continuous refinement meetings with stakeholders” or “sprint review meetings”. The attributes of the metaclass Agile RE Pattern are: name, context, tag, problem summary, usage description, example and template.

In terms of RE, the agile RE pattern can be categorized by an agile RE activity (specified by name and stage). Agile RE activities (see **Table 2**) are slightly different compared to the activities known from established RE (elicitation, analysis and negotiation and validation [21]). Agile RE is carried out in an iterative

Table 2. Activities in agile RE.

Agile RE activity	Description
Discovery	Eliciting new ideas/requirements
Refinement	Clarifying and analyzing new ideas/requirements
Prioritization	Measuring the value that the development will add to the system
Review	Checking if the requirement is implemented in the manner to deliver value
Documentation	Capturing discussion and decisions around the requirement

manner and the activities are not split into phases. In comparison to established RE, they can take place at the same time.

The metaclass Requirement shows how requirements are described in agile RE. This is an external class of the agile RE metamodel since this metamodel provides generic concepts that allow modeling the way people work and collaborate in terms of agile RE. The agile RE metamodel does not aim to give details of the specific shape of a requirement in an agile environment. Therefore, there are other approaches such as those in [22], or [23].

4.2. Additional Views on the Agile RE Metamodel

The human-centric view plays an important role in relation to agile RE. To this end, we present additional views on the agile RE metamodel in order to detail the human aspects it involves.

4.2.1. Metaclass User

The user is a person who directly interacts with the product. As product development is concerned, it is necessary to understand the needs of the user in order to develop a product, which is valuable to the target group. From a HCD perspective, the analysis of the target group is carried out by means of conducting user research. To this end, the user should be modeled with at least some basic information that is listed in **Figure 4** as enumerations.

The personal information allows creating a picture of a concrete person with the aim to avoid self-referential design during product development. Moreover, using behavior helps people to understand how one specific individual interacts with the product. The description of behavior constitutes the picture of a

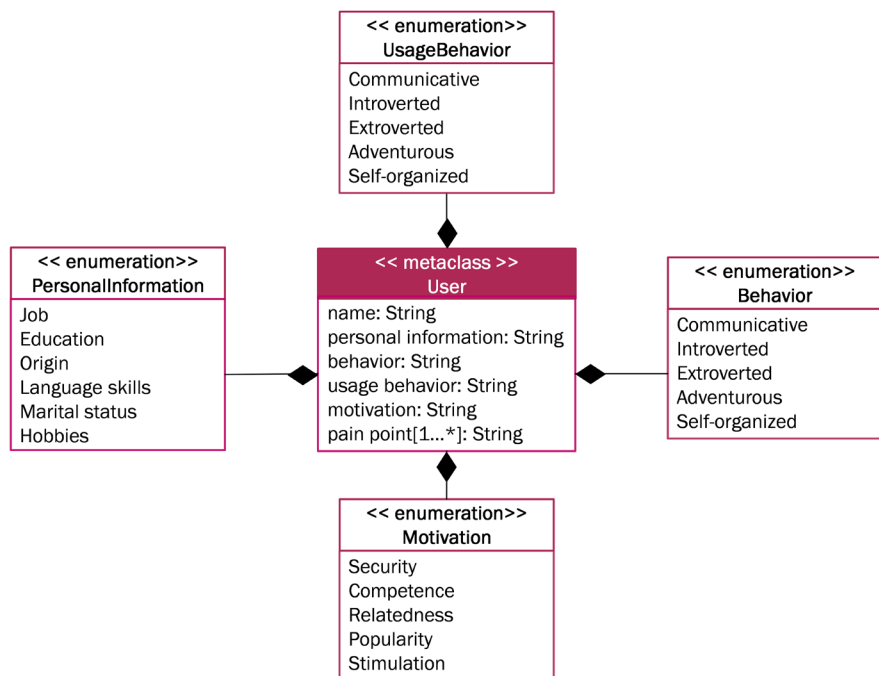


Figure 4. Detailed view on metaclass <<User>>.

concrete user and enables project members to generate an empathetic focus on his/her needs and values.

4.2.2. Metaclass Stakeholder

The stakeholder is an individual or organization having a right, share, claim or interest in a system or in the characteristics that meet their needs and expectations [4]. **Figure 5** highlights some details that make the definition of stakeholders more clear.

4.2.3. Metaclass Agile Team

The metaclass Agile Team details the composition of the product development team who actually work on creating a product. Therefore, the enumeration role (see **Figure 6**) explains the roles an agile team should entail.

4.3. Agile RE Problems and Agile RE Patterns

We have conducted an empirical evaluation through a study with 26 experts from industry where we have identified the most important problems in agile RE [15]. In sum, 6 out of the 20 problems identified have been defined as key problems (see **Table 3**). In light of this, we recommend some options to address such problems by means of agile RE techniques.

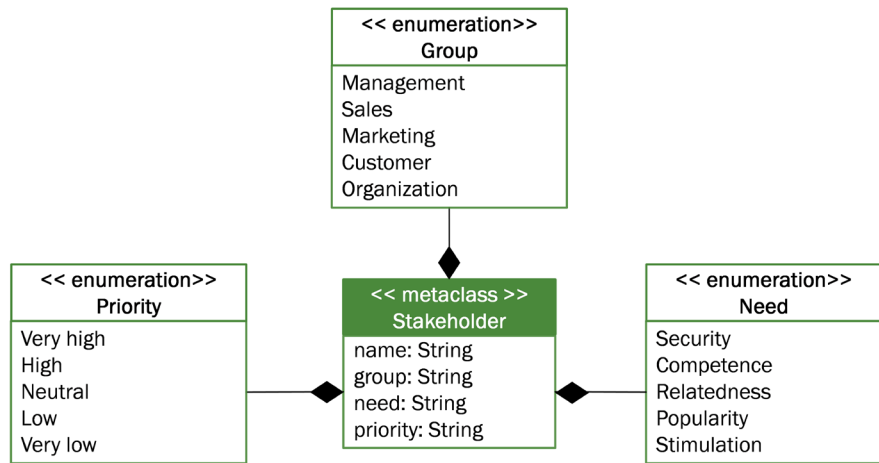


Figure 5. Detailed view on metaclass <<Stakeholder>>.

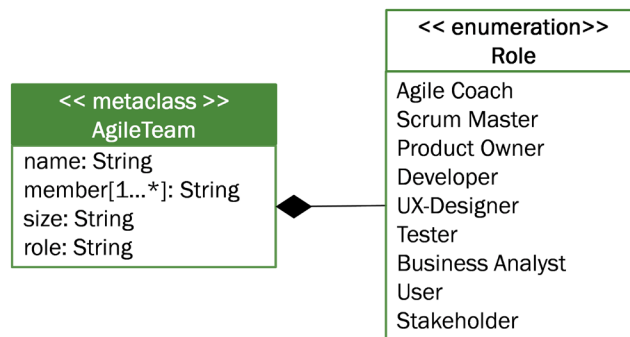


Figure 6. Detailed view on metaclass <<AgileTeam>>.

Table 3. Key problems in agile RE [15].

Problems ID	Description of key problems in agile RE
P1	In ASD, functional or technical dependencies with other teams are a challenge because a considerable coordination effort is required.
P2	In ASD, it is a challenge that stakeholders understand that the development team can make independent (detailed) decisions.
P3	In ASD, it is a challenge not to lose sight of the big picture during the implementation of complex requirements.
P4	In ASD, continuous management of requirements is a challenge since not all of them are fixed at the beginning and they may change over the course of the project.
P5	In ASD, it is a challenge to work out user requirements and quality of use in cooperation with direct users (end users) of the system.
P6	In ASD, it is a challenge to involve stakeholders throughout the whole development process in regular iterations, so that system development will succeed.

The combination of an agile RE problem and the corresponding solution (agile techniques) builds the agile RE pattern. An agile RE pattern is composed of both dynamic and static parts (see **Figure 7**). On the one hand, the agile technique Sprint Planning [24] is an activity and therefore represents the dynamic part. On the other hand, the agile technique User Story [25] is an artifact and represents the static part of an agile RE pattern.

5. Tailoring an Agile RE Domain Specific Model in Industry

The agile RE metamodel can be tailored to domain specific models in order to be applicable to industry. In the following subsections, we will present two cases in which we have built instances of the metamodel. The first process model (case a) is carried out in an environment that uses Scrum for the development of a web application for E-government, with the aim to solve the problem of discovering services (E-government services). In contrast, the second process model (case b) describes an environment that uses Kanban for the relaunch of an internet-based newspaper portal. We have decided to show these two cases, because they demonstrate the adaptability of the agile RE metamodel in terms of tailoring domain specific models for RE based on different agile methodologies.

5.1. Case a: Tailoring in Scrum

Scrum [3] is an agile methodology that can be classified as a time-box approach comprising a set of roles, meetings and artifacts. Since Scrum is described as a framework where people can address complex problems, we need to adapt the framework to the specific context of a project. This case presents an instantiation of the agile RE metamodel in a project using NDT-Agile (based on Scrum and Earned Value Management) to create a web application for e-government.

5.1.1. Background of the Project

Junta de Andalucía is the name of the regional government of the Spanish region

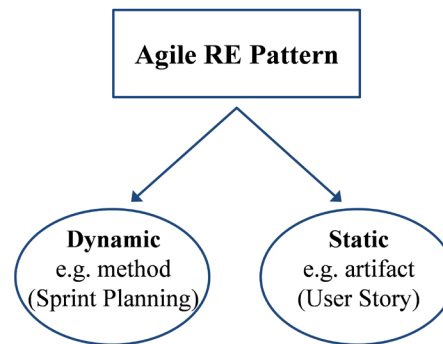


Figure 7. Agile RE patterns are composed of dynamic and static parts.

of Andalusia, in the south of Spain, and the body that has developed this project. The Information and Communication Technologies (ICT) Department of the Ministry of Culture has been in charge of leading the project in liaison with other departments belonging to the regional government of Andalusia.

In between 2008-2012, the Ministry of Culture transformed itself into an organization capable to operate under SOA paradigm [26]. In 2012, they started to adopt agile methodologies [27] [28], but there was a lack: the integration of an activity in the early stages of ASD executing the process of discovering services that, simultaneously, could cover the requirements represented by User Stories.

The next subsections will provide a description of the agile RE metamodel tailored within a public administration so as to solve an agile RE problem regarding discovering services in early stages of ASD. In particular, discovering what services exist in the Services Portfolio, covering the User Story proposed for a new web application. This problem is related to software reusability in early stages of the lifecycle development.

5.1.2. Instantiation into a Scrum-Based Process Model

To face up this drawback, it has been essential to identify all entities, resources and stakeholders in the organizations that are affected by this problem and clarify the relationships among them.

We need several steps to instantiate the proposed agile RE metamodel. First, it is necessary to match up the entities of the metamodel with the real elements of the organization. For this purpose, **Table 4** shows the relationship between each entity and the corresponding element. The metaclasses Organizational Environment and Impact are not presented in this table because they are filled at runtime. Besides, we leave out the metaclass Requirement, since it is external to the approach (see Section 4.1).

As described before, the metaclass Organizational Environment defines the context in which the process model is used. The organizational environment is composed of the metaclasses User, Methodology and Domain (see **Figure 3** and **Table 4**). The instances of the metaclass Agile REProblem and Agile REPattern are described in the form of an agile RE pattern. The pattern Discovering services that cover an agile requirement presented in **Table 5** is used for the agile RE activities discovery and refinement.

Table 4. Instantiation of the agile RE metamodel to process model (case a).

Metaclass	Instantiation for case a
System	The system comprises all information systems involved in these web applications together with the Services Portfolio that belong to the organization. The IT infrastructure is a key part of the system, too.
Domain	The domain of this problem is related to the e-government services and the integration into the ASD development lifecycle.
Context of use	The context of use is the Ministry of Culture that delivers e-government services by means of web applications using agile software developments. The main users of these applications are citizens and public employees.
User	The users are citizens and public employees as well as the IT responsible or even the agile team.
Stakeholder	The stakeholders are people in charge of the different departments (ministry, IT or business).
Agile team	A team belongs to the IT service made up by a Scrum Master and agile developers together with a Product Owner that represents the different interests of stakeholders (the deputy head of the IT department).
Methodology	The methodology followed is NDT-Agile (based on Scrum and Earned Value Management).
Agile RE problem	Technical or functional dependencies to other teams.
Agile RE pattern	Discovering services that cover an agile requirement (see Table 5).
Agile RE activity	Identification and refinement.

Table 5. Pattern “Discovering services that cover an agile requirement”.

Attribute	Description
Pattern name	Discovering services that cover an agile requirement.
Context	Detecting existing services, which can be found in the service portfolio of the organization, that cover user stories functionality identified by the agile team. This pattern is used in early stages of a sprint, directly after the Sprint Planning (see Figure 8) and it is necessary for the software reutilization. The identification of existing services is a very difficult task since the services area is not connected with the agile techniques that are used in the requirements elicitation phase.
Tag	Discovery, refinement and methods. In ASD, functional or technical dependencies with other teams are a challenge because a considerable coordination effort is required (P1, Table 3).
Problem summary	<ul style="list-style-type: none"> • Detection of services in early stages of ASD. • Identification of existing services in the services portfolio that cover the user story proposed for a new web application. • Reusability of software in early stages of the software development lifecycle.
Usage description	<p>It applies the process named DS4aRE (“Discovering services for agile requirements”) that consists of six steps:</p> <ol style="list-style-type: none"> 1) Indexing the Services Portfolio. 2) Formalizing the agile requirement (see Figure 9). 3) Transforming this formalization into a query. 4) Launching this query against the index. 5) Preparing the result of the query. 6) Obtaining the Candidate Services (<i>i.e.</i> a Service that is inside the Services Portfolio and would cover an agile requirement).

Continued

Example	The Ministry needs to develop a new web application to comply with Law 39/2015 that obliges to make any notification by telematics means. The Ministry needs to know how many services of the e-government Services Portfolio could be used in this application to cover the User Stories identified by the agile team.
Template	DS4aRE process (see Figure 10).

Figure 8 shows the moment in which the agile team must execute the activity known as Discovering Candidate Services, which has been integrated into the agile methodology in the Ministry of Culture.

Figure 9 details the elements used during the discovery of Candidate Services (see **Figure 8**). In particular, it models the necessary entities to represent a User Story and other information about functional and non-functional requirements, users, stakeholders and agile team after finishing the sprint planning in NDT-Agile.

Moreover, **Figure 10** shows the DS4aRE (Discovering services for agile requirements) process that solves the RE problem exposed in **Table 5**.

5.2. Case b: Instantiation within a Kanban-Based Environment

Kanban [2] is an agile methodology whose main attribute focuses on continuous improvement. The application of Kanban starts with making a workflow visible and proceeds with continuous improvement of the existing process model. Compared to the time-box approach in Scrum, Kanban is based on a continuous flow model. In the following case, we will present an instantiation of the agile RE metamodel into a Kanban-based process model for developing an internet-based newspaper portal.

5.2.1. Background of the Project

The project was carried out in a medium-sized IT company, located in Germany [29], specialized in e-commerce, mobile apps and SAAS tools. The project team consisted of twelve members (one team leader, one project manager, two visual designers, two UX experts and six developers). The aim was the relaunch of an internet-based newspaper portal within six months in 2013/2014.

5.2.2. Instantiation into a Kanban-Based Process Model

In the following paragraphs, we will provide an instantiation of the metamodel (**Figure 3**) and we will discuss the way of working within the scope of case b. **Table 5** shows the instances of the metaclasses in terms of the applied process model. The metaclasses `OrganizationalEnvironment` and `Impact` are not presented in this table since they are filled during runtime. Moreover, we leave out the metaclass `Requirement`, since it is external to the approach (see Section 4.1).

As described before, the metaclass `OrganizationalEnvironment` defines the context where the process model is used. The organizational environment is a composition of the metaclasses `User`, `Methodology` and `Domain` (see **Figure 3**

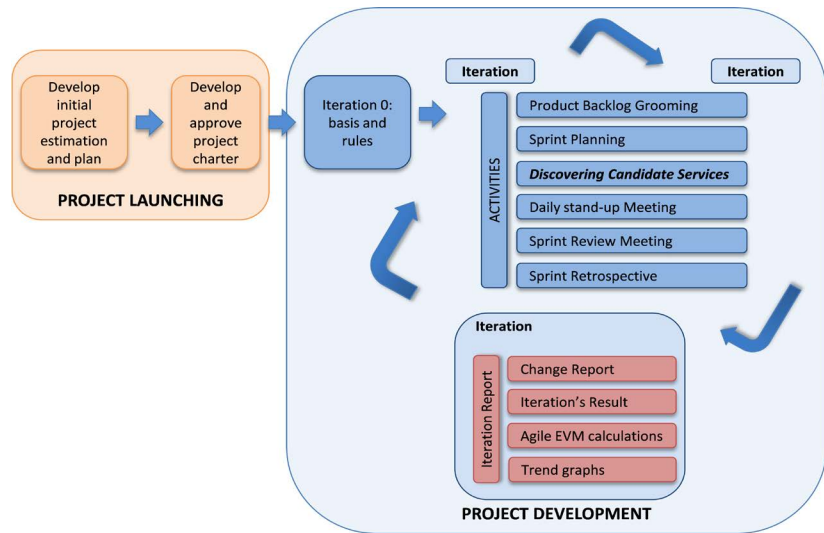


Figure 8. Activity “discovering candidate services”.

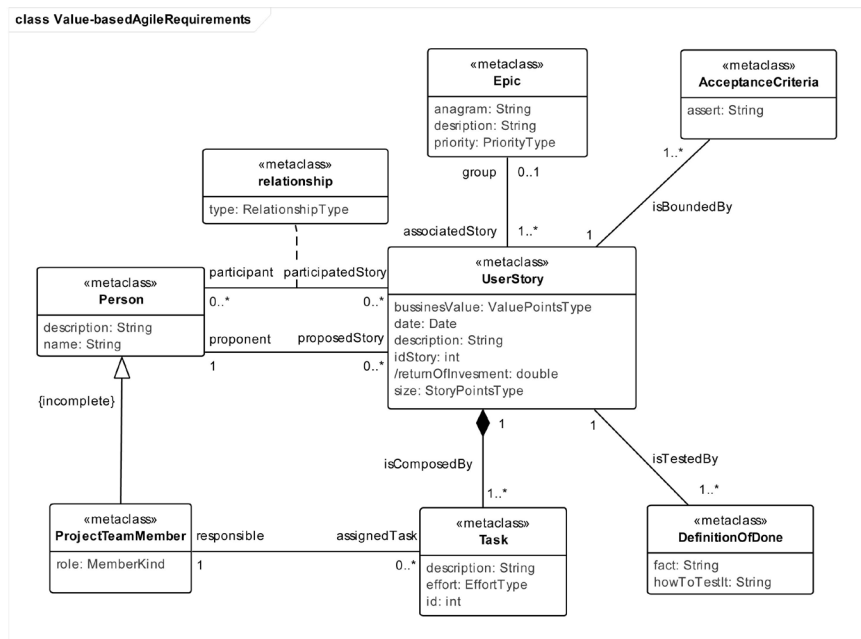


Figure 9. Value-based agile requirements metamodel for the agile RE pattern “Discovering services that cover the agile requirement”.



Figure 10. DS4aRE process for pattern “Discovering services that cover the agile requirement”.

and Table 6). The instances of the metaclass Agile RE Problem and Agile RE Pattern are described in the form of an agile RE pattern. The pattern Continuous management of requirements by means of tools, presented in Table 7, is utilized for the agile RE activities discovery and refinement.

Table 6. Instantiation of agile RE metamodel to process model demonstrated in case b.

Metaclass	Instantiation for case b
System	The system to be developed is an internet-based newspaper portal.
Domain	The domain where the system is used can be classified as e-commerce in the publishing sector.
Context of use	The main task of users consists in retrieving news, that is to say, they want to be informed of contemporary issues. Therefore, the first touch point of users is the home page where they can browse through the teasers and select the interesting ones for further reading. In most of the cases they use equipment comprising a mobile platform (tablet or smartphone). Both the physical environment and the social environment of users depend on the situation in which they use the system. For instance, they can use the system in public transport that may be loud and crowded, or they can use it in their gardens where the sun is shining on the display.
User	The user of the system is an experienced web user who is interested in news. Printed media is too slow for her/him and s/he appreciates consuming videos and additional interactive content.
Stakeholder	There are a lot of different stakeholders involved within the development process of the system (e.g. management, investors or shareholders).
Agile team	The agile team consists of twelve members: one team leader, one project manager, two visual designers, two UX experts and six developers.
Methodology	The methodology integrates Kanban with HCD.
Agile RE problem	Continuous requirements management.
Agile RE pattern	Continuous management of the requirements by means of tools (see Table 7).
Agile RE activity	Discovery and refinement.

Table 7. Pattern “Continuous management of requirements by means of tools”.

Attribute	Description
Pattern name	Continuous management of requirements by means of tools
Context	Established RE approaches fit into sequential approaches to software development. On the contrary, ASD is used to enhance the ability to deal with changing requirements over the course of time. Agile techniques for continuous requirements management need to be implemented in order to ensure traceability of requirements.
Tag	Discovery, refinement and methods.
Problem summary	In ASD, continuous management of requirements is a challenge since not all of them are fixed at the beginning and they may change over the course of the project (P4, Table 3). <ul style="list-style-type: none"> • Providing a positive UX to the user. • Carrying out a release evaluation continuously. • Not interrupting the workflow due to scheduling testing activities and organization. • Reducing costs for long-term UX testing.
Usage description	The workflow of the system development is visualized by means of Kanban boards for different types of tasks (e.g. UX design, development or operation). One requirement can be tracked along the workflow and its evolution is managed through the whole development process (see Figure 11). Organizing the management of requirements by means of Kanban boards implies a continuous management and tracking of changes.

Continued

Example	<p>Figure 11 presents the interaction of two Kanban boards in case b. We use a third one for operation that was placed on the right hand side of the development board. Kanban boards represent the workflow from design through development of the internet-based newspaper portal. One task from the design board might be split into more than one task on the development or the operation board. The aim of this procedure is to obtain a continuous flow within the board and among the boards. For working with multiple Kanban boards the project team use tool support by means of JIRA from Atlassian, displayed on several screens on the wall.</p>
Template	See Figure 11 .

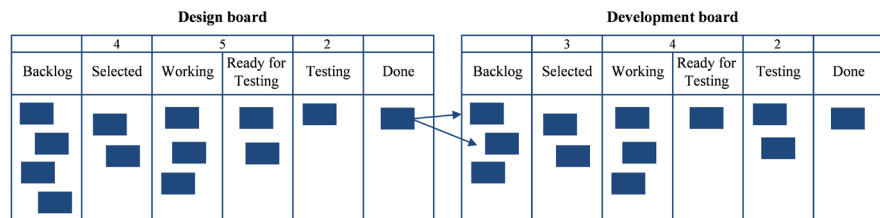


Figure 11. Example of multiple Kanban boards for pattern “Continuous management of requirements by means of tools”.

It is worth mentioning that more than one of the presented agile RE problems occurred along the development of project case b. For instance, it was a problem not to lose sight of the big picture during the implementation of complex requirements since the development with Kanban is focused on small tasks. Therefore, we integrate HCD activities by means of a release evaluation [29].

6. Discussion and Limitations

One of the most important benefits of the proposed metamodel for agile RE is cross-functionality. Agile methodologies are based on cross-functional collaboration of people from different disciplines (e.g. programming, test or UX design). It is essential to consider all the disciplines while creating a process model for agile RE, for becoming a value-driven organization. The proposed metamodel for agile RE supports the understanding of this complexity and enables companies to become clear about the influencing parameters. Furthermore, researchers as well as practitioners can use the metamodel for evaluating their existing process models for agile RE aiming to improve them.

The application of the agile RE metamodel allows us to reflect on the agile RE problems that happened in the presented cases (see Section 5.1.2 and Section 5.2.2) as well as to find out appropriate solutions for handling them.

Once these two cases are analyzed, we can figure out some differences in terms of the elaboration of information related to specific stereotypes. Basically, the context of use is deeper explored and more elaborated in case b (see Section 5.2.2). This occurs because of the integration of HCD, as in HCD the context of use is meaningful in terms of user research. Taking this as an example, we can discuss whether it makes sense to put some effort in the analysis of the context of

use in case a, and discuss what impact this analysis might have on the value delivery. Besides, they also differ in the manner of how people are related among them. In case a, for instance, the user is part of the agile team, since some members of the agile team use the system, such as citizens or employers of the Ministry of Culture. Finally, another difference can be observed in the relation between the methodology and the agile RE pattern. In case b, the agile RE pattern is part of the methodology, since the agile technique Kanban board is rooted in the Kanban methodology, whereas the agile technique in case a (DS4aRE process) is a new technique independent of a methodology.

Nonetheless, the analyzed cases in this work present some shortcomings in terms of the analysis of the organizational environment. We have to state that we were not able to gather all the data the metamodel would have the capacity to include due to time spent when the case studies were carried out. The final version of the metamodel was created in July 2017, whereas the case studies were conducted before that time.

As opposed to this, both cases have confirmed that domain specific models allow us to receive a detailed view on complex organizational environments. This detailed view enables us to identify problems and impediments that have to do with people, processes or tools. Furthermore, it helps us to compare specific excerpts of different organizational environments with the aim to reflect on all the aspects that run properly and should be shared.

The use of patterns is a reasonable approach since agile RE is a problem-solving approach to ASD. The usage of patterns is widely spread in the field of software engineering [30]. Most of the proposals rely on the approach by Alexander *et al.* [31]. The concept of agile RE patterns presented in the agile RE metamodel (see **Figure 3**) is also based on Alexander *et al.*'s [31] understanding for the use of patterns.

In light of this, the combination of agile RE problems and their corresponding solutions can be inferred to new agile RE patterns that can make easy sharing the experience with the community. The gathered knowledge can encourage other people to solve similar individualized agile RE problems to those presented in this work.

Nevertheless, there might be some limitations to our research. We have identified the agile RE problems and agile RE patterns by empirical evaluation. It could be argued that there are some limitations in terms of the general application of the results due to the usage of an expert validation. However, the experts work for different companies and therefore they have different levels of experience in the agile transformation of an organization. Additionally, the identified key problems (**Table 3**) only give guidance for the optimization of existing process models for agile RE in industry and do not claim to be fully complete.

Deriving a process model for agile RE based on the proposed agile RE metamodel might still be a challenge in some organizations due to the complexity of the environment (e.g. scaled organization or other teams working with sequential approaches like waterfall model). Nonetheless, the metamodel reduces this

complexity by visualizing the influential parameters that can be changed according to the explicit view.

7. Conclusion and Future Research

This paper presents the agile RE metamodel that builds an abstraction layer about the variety of existing agile RE process models used in industry. The metamodel for agile RE helps us to describe key concepts of agile RE management. This gives us a common definition for this research area and supports us in the development of a unified language. Besides, we have demonstrated how we can build instances of the metamodel in order to provide process models for industry, concretely in Scrum and Kanban environments. We have presented two cases that analyze an instantiation of the metamodel: in case a, an agile RE process model for NDT-Agile (based on Scrum and Earned Value Management) is used for creating a web application that enables the Ministry of Culture to comply with Law 39/2015; in case b, a Kaban-based agile RE process model is utilized for developing an internet-based newspaper portal.

We have contributed to the software development body of knowledge by providing a metamodel for agile RE. The metamodel has several implications for both, researchers and practitioners. On the one hand, the metamodel can be used to model new process models for agile RE that are strongly focused on value delivery and therefore can be classified as value-driven process models. On the other hand, existing process models can be evaluated as well as improved by the metamodel.

Future research may specifically deal with providing practitioners and researches with tool support so that they can improve their agile RE process models based on the proposed agile RE metamodel. We would like to encourage them to improve their agile RE process models. To this end, we are currently working on a web application that provides guidance for solving agile RE problems by choosing the proper agile technique. The objective is to share knowledge regarding agile RE by means of agile RE patterns. The web application is a knowledge base concerning the most common agile RE problems in industry and appropriate solutions.

Acknowledgements

We would like to thank Dr. Nora Koch for her support in the improvement of the agile RE metamodel and Dominique Winter for his support in carrying out the empirical evaluation related to agile RE problems. We are also very grateful to Dr. Francisco José Domínguez Mayo for the valuable discussion concerning the agile RE metamodel.

Supported

This research has been supported by the Pololas project (TIN2016-76956-C3-2-R) of the Spanish Ministry of Economy and Competitiveness.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Schwaber, K. (2004) *Agile Project Management with Scrum*, Microsoft. Pearson Education, London.
- [2] Anderson, D.J. (2010) *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press, Washington DC.
- [3] Beck, K. (2000) *Extreme Programming Explained: Embrace Change*. Addison-Wesley, Boston.
- [4] International Organization for Standardization (2010) ISO 9241-210:2010—Ergonomics of Human-System Interaction—Part 210: Human-Centred Design for Interactive Systems.
- [5] Schön, E.-M., Thomaschewski, J. and Escalona, M.J. (2017) Agile Requirements Engineering: A Systematic Literature Review. *Computer Standards & Interfaces*, **49**, 79-91. <https://doi.org/10.1016/j.csi.2016.08.011>
- [6] Inayat, I., Salim, S.S., Marczak, S., Daneva, M. and Shamshirband, S. (2015) A Systematic Literature Review on Agile Requirements Engineering Practices and Challenges. *Computers in Human Behavior*, **51**, 915-929. <https://doi.org/10.1016/j.chb.2014.10.046>
- [7] Heikkilä, V.T., Damian, D., Lassenius, C. and Paasivaara, M. (2015) A Mapping Study on Requirements Engineering in Agile Software Development. 2015 *41st Euro-micro Conference on Software Engineering and Advanced Applications*, Funchal, 26-28 August 2015, 199-207. <https://doi.org/10.1109/SEAA.2015.70>
- [8] Medeiros, J., Alves, D.C.P., Vasconcelos, A., Silva, C. and Wanderley, E. (2015) Requirements Engineering in Agile Projects: A Systematic Mapping Based in Evidences of Industry. *CibSE 460-473, XVIII Ibero-American Conference on Software Engineering (CIBSE)*, Lima, 22-24 April 2015, 460-473.
- [9] Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., *et al.* (2001) Manifesto for Agile Software Development. <https://agilemanifesto.org/>
- [10] Memmel, T., Gundelsweiler, F. and Reiterer, H. (2007) Agile Human-Centered Software Engineering. *Conference. Proceedings of the 21st British HCI Group Annual Conference on HCI 2007*, Lancaster, 3-7 September 2007, 167-175.
- [11] Kautz, K. (2010) Participatory Design Activities and Agile Software Development. *IFIP Working Conference on Human Benefit through the Diffusion of Information Systems Design Science Research*, Perth, 30 March-1 April 2010, 303-316. https://doi.org/10.1007/978-3-642-12113-5_18
- [12] Maguire, M. (2013) Using Human Factors Standards to Support User Experience and Agile Design. *International Conference on Universal Access in Human-Computer Interaction*, Las Vegas, NV, 21-26 July 2013, 185-194. https://doi.org/10.1007/978-3-642-39188-0_20
- [13] Rivero, J.M., Grigera, J., Rossi, G., Robles Luna, E., Montero, F. and Gaedke, M. (2014) Mockup-Driven Development: Providing Agile Support for Model-Driven Web Engineering. *Information and Software Technology*, **56**, 670-687. <https://doi.org/10.1016/j.infsof.2014.01.011>

- [14] Olsson, H.H. and Bosch, J. (2015) Towards Continuous Customer Validation: A Conceptual Model for Combining Qualitative Customer Feedback with Quantitative Customer Observation. LNBIP 210, *International Conference of Software Business*, Braga, 10-12 June 2015, 154-166.
https://doi.org/10.1007/978-3-319-19593-3_13
- [15] Schön, E.-M., Winter, D., Escalona, M.J. and Thomaschewski, J. (2017) Key Challenges in Agile Requirements Engineering. XP 2017, *International Conference on Agile Software Development*, Cologne, 21-25 May 2017, 37-51.
- [16] Schön, E.-M. (2016) A Methodology for Agile Requirements Engineering Based on a Pattern Approach. *Proceedings of CAiSE 2016 Doctoral Consortium*, Ljubljana, 1-8.
- [17] Schön, E.-M., Thomaschewski, J. and Escalona, M.J. (2017) Identifying Agile Requirements Engineering Patterns in Industry. In: *Proceedings of the 22nd European Conference on Pattern Languages of Programs*, ACM Press, New York, 1-10.
- [18] Rolland, C. (1993) Modeling the Requirements Engineering Process. *3rd European-Japanese Seminar on Information Modeling and Knowledge Bases Budapest*, Hungary, 1-12.
- [19] Schön, E.-M. (2017) A Framework for Modeling and Improving Agile Requirements Engineering. Ph.D. Thesis, University of Seville, Sevilla.
- [20] Evans, E.J. (2003) *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison Wesley, Boston.
- [21] Sommerville, I. and Sawyer, P. (1997) *Requirements Engineering: A Good Practice Guide*. John Wiley & Sons, Inc., New York.
- [22] Sedeño, J. (2017) Una propuesta para el descubrimiento sistemático de servicios en fases tempranas de desarrollos ágiles. Ph.D. Thesis, University of Seville, Seville.
- [23] Escalona, M.J. and Aragón, G. (2008) NDT. A Model-Driven Approach for Web Requirements. *IEEE Transactions on Software Engineering*, **34**, 377-394.
<https://doi.org/10.1109/TSE.2008.27>
- [24] Schwaber, K. and Sutherland, J. (2016) *The Definitive Guide to Scrum*.
- [25] Cohn, M. (2004) *User Stories Applied: For Agile Software Development*.
- [26] Sedeño, J., Torrecilla-Salinas, C.J., Escalona, M.J. and Mejías, M. (2014) An Approach to Transform Public Administration into SOA-Based Organizations. *10th International Conference on Web Information Systems and Technologies*, Barcelona, 3-5 April 2014, 135-142.
- [27] Torrecilla-Salinas, C.J., Sedeño, J., Escalona, M.J. and Mejías, M. (2013) Using an Agile Framework to Deliver e-Government Services in Public Administrations. *21st Annual Software Quality Management*, London, 2-5 September 2013, 167-175.
- [28] Torrecilla-Salinas, C.J., Sedeño, J., Escalona, M.J. and Mejías, M. (2015) Estimating, Planning and Managing Agile Web Development Projects under a Value-Based Perspective. *Information and Software Technology*, **61**, 124-144.
<https://doi.org/10.1016/j.infsof.2015.01.006>
- [29] Schön, E.-M., Winter, D., Uhlenbrok, J., Escalona, M.J. and Thomaschewski, J. (2016) Enterprise Experience into the Integration of Human-Centered Design and Kanban. *Proceedings of the 11th International Joint Conference on Software Technologies*, Lisbon, 24-26 July 2016, 133-140.
<https://doi.org/10.5220/0005942601330140>
- [30] Henninger, S. and Corrêa, V. (2007) Software Pattern Communities: Current Practices and Challenges. *Proceedings of 14th Conference Pattern Languages Programs*,

Monticello, 5-8 September 2007, 1-19.

- [31] Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I. and Angel, S. (1977) *A Pattern Language: Towns, Buildings, Construction* (Center for Environmental Structure). Oxford University Press, Oxford.