An Agile approach to CMMI-DEV levels 4 and 5 in Web development projects. [12]

C.J. Torrecilla-Salinas *carlos.torrecilla@iwt2.org*
Department of Computer Languages and Systems. University of Seville (Spain)

J. Sedeño *jorge.sedeno@iwt2.org*
Agencia Andaluza de Instituciones Culturales. Seville (Spain)

M.J. Escalona *mjescalona@us.es*
Department of Computer Languages and Systems. University of Seville (Spain)

M. Mejías *risoto@us.es*
Department of Computer Languages and Systems. University of Seville (Spain)

Abstract

CMMI (Capability Maturity Model-Integration) model proposes a set of process areas, including suggested practices, with the aim of helping organizations to improve the quality of their products and processes. It is commonly accepted that as an organization progresses through the different levels of CMMI, the quality of its development might improve as well as the overhead of the development process, impeding it to quickly adapt to customers or partners changing needs. Besides, Agile practices allow quick adaptation and early delivery of business value. The specificity of Web environments makes them suitable for Agile approaches. However, as quality requirements for Web systems increase, a combination of Agile practices allowing organizations to achieve higher levels of CMMI-DEV with a limited process overhead can be very interesting to organizations that aim to keep adaptability. This way, they might strengthen their development processes in order to produce high quality results. This paper presents a gap analysis between the most used Agile practices (Scrum and XP) as well as a mapping proposal, including ad-hoc modifications and other Agile practices, to achieve all CMMI-DEV level 4 and 5 specific goals. To conclude, it drafts relevant conclusions and proposes future lines of research.

Keywords: Agile, Scrum, eXtreme Programming, Web Engineering, CMMI, Software Engineering.

1. Introduction

During the last decade, after the appearance of the "Agile manifesto" [2], Agile methods, practices and techniques have established themselves as a valid alternative for systems and software development [1]. Agile principles, as exposed in the aforementioned manifesto, include the adaptation to user changing needs, even late in the development process.

Additionally, Web Engineering studies Web systems, as those systems to be published and consumed in the Web [8]. Web development projects have more different needs and characteristics than other type of software development project producing desktop or embedded software [8, 28, 32]. Some of these characteristics are: different navigational structure, increased security and maintainability requirements, reduction of feedback loop with final users, reduction on features delivery and quick adaptation to user changing needs.

[12] This paper presents the authors' views, which do not necessarily reflect those of their employers.

It is worth mentioning that since the appearance of the “Agile Manifesto”, Agile has been embraced by more and more organizations including some of the major actors like Microsoft, Amazon or Yahoo. This trend is similar in Web development projects [1] concerning not only companies, but also research groups [23].

CMMI-DEV (Capability Maturity Model Integration – Development) belongs to CMMI maturity models family, which proposes different models that will allow organizations to improve their development processes. The progress through the different maturity models proposed by CMMI is linked to improvement on product quality and customer satisfaction [14]. Nevertheless, there are sometimes reluctances to apply this model, for example, in small organizations [37]. The relations between Agile and CMMI, and between both and Web Engineering have been object of study during the past years [35, 40], but a common trend in the published work is the lack of analysis of CMMI proposed practices for higher levels (level 4 and 5). These levels are dealing with organizational matters and continuous improvement, among other topics. A proposal that might identify Agile practices to cover CMMI-DEV specific goals will help to overcome the identified reluctances to implement CMMI. It will also allow organizations to institutionalize Agile by achieving CMMI-DEV higher maturity levels, without increasing project overhead and keeping the necessary agility to adapt to future changes.

As it is known, Scrum and eXtreme Programming (XP) are the most used approaches [27, 44] from Agile family, being also quite common to find a combined implementation of both of them. This is the reason why any approach to achieve CMMI goals might first take into account Scrum and XP practices. Finally, it is important to mention that this paper might be considered as the continuation of our work [42, 43] regarding the relations among Agile, CMMI and Web Engineering for maturity levels 2 and 3. Based on the foregoing, our paper tends to reach the following goals:

1. Present a gap analysis between CMMI-DEV maturity levels 4 and 5 goals and proposed Scrum and XP practices.
2. Recommend some Agile techniques to fill in the identified gap between CMMI maturity levels 4 and 5 and Scrum/XP.
3. Present conclusions and suggest further lines of research.

For this purpose, it is organized as follows: after this introduction, Section 2 will present the research questions and method. Afterwards, Section 3 will introduce the background associated with CMMI-DEV, Agile and Web Engineering. Section 4 will summarize the identified related works. Then, Section 5 will explain the gap analysis and the mapping proposal, and finally, Section 6 will draft the main conclusions of the paper and will recommend further lines of research.

2. Research questions and research method

This section will briefly introduce the research questions and the research approach followed both to map the different Scrum and XP practices to the different CMMI maturity levels 4 and 5 specific goals, and to distinguish existing suitable Agile techniques that might fill in the identified gaps. Our research question is linked to this more generic question: “*Is an Agile approach to CMMI levels 4 and 5 feasible for Web development organizations?*” As it can be observed, this is a very extensive and general question. Thus, in order to carry out our task, we focused on the three more detailed and specific research questions that are listed below:

- *RQ1:* What is the gap, if any, between Scrum and XP practices and CMMI levels 4 and 5 goals?
- *RQ2:* How CMMI levels 4 and 5 goals can be related at the same time to Web specificities and to Scrum/XP practices?
- *RQ3:* In case that a gap between Scrum/XP practices and CMMI levels 4 and 5 goals exists, are there any other Agile techniques that can cover such a gap?

Our approach started with a gap analysis between Scrum and XP and CMMI-DEV levels 4 and 5 goals, in order to answer the aforementioned research questions. To carry on with that study, we first studied in detail each goal description and its identified practices as provided in CMMI-DEV 1.3 standard [7]. Once they were well understood, both Scrum and XP “primary sources” [3, 39] were analyzed in detail to identify whether one of the proposed practices could achieve the described goals. Other existing and related mapping exercises or gap analyses were used as complementary sources to support our decision, and they will be described later on. If a clear set of practices from the methodology were identified, the CMMI goal would be marked as completely covered. If just a few practices were related, but it was not clear that the goals were achieved by implementing the methodology, we stated that the goal would be just partially covered. Finally, if no associated practice were found in the “primary sources”, the goal would be marked as not covered. All those goals that we marked as not covered became the “gap” between Scrum/XP and the analyzed CMMI-DEV levels. The next step consisted then in identifying, from the analysis of the existing Agile literature, other Agile practices, techniques or methods that could help to achieve the identified goal. If existing practices were identified, we would propose them as relevant for our purpose. If no suitable practices were found, we would propose an “ad-hoc” extension to Scrum or XP. Every extension will be designed keeping the Agile principles in mind, for example, not to increase process overhead in excess or keep the quick delivery of value in mind, among others.

3. Background

In this Section we will present a high-level background of the research related topics (CMMI, Agile and Web Engineering), in order to provide a context for the later sections.

CMMI-DEV

CMMI proposes a different set of models with the aim of allowing organizations to develop better processes [7]. Within the CMMI framework, CMMI-DEV1.3 is today’s last version of the maturity model focused on software development [7]. CMMI-DEV includes twenty-two process areas, each one with different goals and practices. As it is known, CMMI models are structured in different levels, which help organizations to improve their development processes. It is also important to know that CMMI model recommends two different representations, providing distinct implementation paths: continuous and staged representations. In our study, we will analyze the staged representation, whose levels are based on the improvement of a particular set of process areas that enables the organization to be prepared for the next one.

Agile methodologies

As explained, Agile is a label that groups a set of methodologies, frameworks, techniques or methods that share a common set of principles and values. Agile methodologies have increased their popularity [4] and are now being successfully used in several projects [6, 36]. Under this group of techniques, Scrum and XP are the two most used approaches [27], and both constituted the starting point of our study. On the one hand, Scrum is a framework for product development that recommends an incremental and iterative approach. On the other hand, Kent Beck and Cynthia Andres [3] proposed XP as a software development methodology that was designed to produce better software with lower cost and in a shorter period of time. It falls into the set of Agile methodologies with a very special focus both on technical excellence and on the development team.

Web Engineering

Web Engineering is the branch of Software Engineering in charge of studying Web systems. As stated, Web systems have special characteristics that differentiate them from the rest of software systems. Several approaches have emerged in Web Engineering, for instance UWE (UML Web Engineering) [19], IFML (Interaction Flow Modeling Language) [25], and NDT (Navigational Development Techniques) [12], among others. Table 1 summarizes Web systems characteristics:

Table 1. Web specific characteristics.

Id	Description	Reference
Wb1	Complex navigational structure	[12, 13]
Wb2	Critical interface requirements (such as unknown users or availability, among others)	[12, 13]
Wb3	Security aspects	[16]
Wb4	Increase on maintenance efficiency, avoiding downtimes	[24]
Wb5	Delivery as soon as possible	[22, 28, 33]
Wb6	Reduction of time-to-market	[22, 28, 33]
Wb7	Adaptation to quick-changing requirements	[22, 28]

Table 1 identifies up to 7 special characteristics of Web projects. Although some of these characteristics can also appear in other type of systems, like real-time or embedded systems, the concurrence of all of them at the same time is normally a singularity of Web developments.

4. Related works

This section is divided in two subsections. The first one offers an overview of the works analyzing the relationship of the different topics studied in this paper (CMMI, Agile and Web Engineering). The second presents a detailed analysis of the existing mappings between Agile and CMMI-DEV levels 4 and 5 process areas performed by other authors.

Related literature

The related works presented in this section were compiled following a Systematic Literature Review (SLR) [40]. Below, we will summarize the aforementioned works, pointing mainly to the identified papers that relate CMMI to Agile in Web contexts. To start, we can find the work by Bougron *et al.* [5], where the authors mapped the goals of CMMI maturity level 3 process areas to the proposed practices of three Agile methods: Scrum, XP and Kanban. The study evaluated which technique, proposed by any of the three different Agile methods, could meet every goal proposing a percentage of coverage. It concluded that the three methods were complementary and able to reach a large number of CMMI level 3 goals. Finally, it provided a case study concerning CMMI level 3 in a particular company using Agile processes to validate the proposal. The paper looked at generic development, without taking into account Web specificity. Nevertheless, it offered useful hints regarding the general compatibility between CMMI and Agile. This work only highlighted CMMI level 3 process areas, and it did not present any proposal to fill in the identified gaps.

Selleri Silva *et al.* [35] studied how feasible using Agile techniques to meet the goals of the CMMI process areas linked to Quality Assurance in its different maturity levels was, by means of defining an Agile Quality Assurance Maturity Model with a similar approach to that of CMMI. They also offered the results of a survey on the analyzed topic. The paper only covered in detail aspects linked to quality of generic development, but not all the remainder elements composing a development project, like project management or engineering, among others.

In the paper by Torrecilla *et al.* [42], there was a theoretical combination of different Agile practices that allowed coping with all specific and generic goals of CMMI maturity level 3, specifically in Web development environments. Besides, practices from different methodologies were evaluated and mapped to the different process areas of CMMI level 3. The work was purely a theoretical gap analysis, without an experiment or case study that could validate the suggested proposal.

Lukasiewicz and Miler [20] presented a model, called Model C-S, that mapped the specific processes of CMMI levels 2 and 3 to Scrum practices. It involved 123 practices, but excluded some of the CMMI process areas associated with the organizational structure (such as those of level 4). Additionally, the paper described which practices were fully, partially or not covered. It suggested that ad-hoc extensions should be utilized for those practices that Scrum standard practices uncovered. It also included a diagnostic questionnaire, a selection algorithm, an application process and a tool, which were further described in the paper. To conclude, the work contained two case studies in order to validate the model.

Torrecilla *et al.* [43] evaluated the feasibility of achieving CMMI maturity level 2 using only Scrum standard practices and techniques for Web development projects. Besides, a state-of-the-art analysis of the question was included, together with a theoretical assessment on whether Scrum techniques could be used or not to achieve the goals of all CMMI level 2 process areas. From the conclusions of that assessment, an extension to Scrum based on other Agile methods (like XP) or ad-hoc modifications was proposed with the aim to fill in the identified gaps.

Another interesting paper is the one of Díaz *et al.* [10]. It studied how Scrum practices could be mapped to certain CMMI maturity level 2 process areas, such as Project Planning (PP), Project Monitoring and Control (PMC) and Requirements Management (REM). It analyzed every specific practice of such process areas, by verifying whether Scrum standard practices could achieve the goal and identify the gaps between both models. Therefore, it presented a case study as a formal assessment of an internal project.

From the theoretical point of view, Marçal *et al.* [21] evaluated whether Scrum standard practices could meet the objectives of particular practices regarding Project Management Process Areas of CMMI maturity levels 2, 3 and 4. It presented a deep theoretical analysis of twenty two of those specific practices and their relation to Scrum techniques, as well as determined if the goals were fully, partially or non-achieved. It did not consider a case study to assess conclusions.

Finally, Paulk [26] explored the practices proposed by XP and CMMI levels 2 and 3 process areas from a theoretical point of view. It concluded that XP could fulfill most level 2 practices and cope with some of level 3 ones. It also stated that XP would be more productive whenever the project size remained small.

To summarize, we can assert that none of the identified works completely mapped Agile to CMMI for levels 4 and 5, considering Web specificities. Thus, this will be the gap that our work will try to fill in.

Gap analysis in the related literature

This section will cope with the relevant works that map different process areas of CMMI levels 4 and 5 to different Scrum and XP practices. Table 2 shows the unique identified work, as it can be seen in the previous section:

Table 2. CMMI-DEV levels 4 and 5 – Scrum/XP mapping relevant works.

Id	Title	Authors	Method	Web?	Level
L4r1	Mapping CMMI Project Management Process Areas to SCRUM Practices	Marcal *et al.*	Scrum	No	4

It is important to note that L4r1 [21] provides an analysis of Scrum practices against CMMI-DEV version 1.2 and does not include Web specificities, as it is only related to general development projects. No work mapping Scrum or XP to CMMI maturity level 5 practices or goals was found. Table 3 displays the results of the gap analysis identified in the highlighted work:

Table 3. CMMI-levels 4 and 5 – Scrum/XP related work gap analysis.

		Related work gap analysis
		L4r1 - Scrum
OPP	OPP-SG 1: Establish Performance Baselines and Models	??
QPM	QPM-SG 1: Prepare for Quantitative Management	--
	QPM-SG 2: Quantitatively Manage the Project	--
CAR	CAR-SG 1: Determine Causes of Selected Outcomes	??
	CAR-SG 2: Address Causes of Selected Outcomes	??
OPM	OPM-SG 1: Manage Business Performance	??
	OPM-SG 2: Select Improvements	??
	OPM-SG 3: Deploy Improvements	??

??: Not analyzed
--: Not covered (no specific practice of the specific goal covered)

Table 3 lets us deduce that the only available work offers a limited analysis, studying simply two of the three specific goals of maturity level 4 and none of level 5. This work only studies project management process areas of CMMI-DEV. In the coming section, we will introduce our own gap analysis, both for Scrum and XP and we will also link the proposed practices to Web specificities for CMMI maturity levels 4 and 5.

5. Gap analysis and mapping proposal.

CMMI-DEV maturity level 4 is called *Quantitatively Managed*. According to CMMI definition, at this level "*the organization and projects establish quantitative objectives for quality and process performance, and use them as criteria in managing projects*". These objectives are based on the different stakeholders needs (customer/user or organization, for instance) and performance is measured and analyzed statistically. The main difference between levels 3 and 4 is the predictability of process performance, as in level 4 processes are controlled through quantitative data. Table 4 shows the two process areas included in CMMI-DEV level 4:

Table 4. CMMI-DEV level 4 process areas.

Process area	**Category**	**Specific goals**	**Specific practices**
Organizational Process Performance (OPP)	Process Management	1	5
Quantitative Project Management (QPM)	Project Management	2	7

CMMI-DEV maturity level 5 is called *Optimizing*. According to CMMI definition, at this level "*an organization continually improves its processes based on a quantitative understanding of its business objectives and performance needs*". At this point, the organization tries to enhance the process performance through incremental improvements, according to established and reviewed execution objectives. Processes are measured using quantitative techniques for these established objectives. Levels 4 and 5 differ in their implementation of the organizational performance by working with data gathered from the different projects. Table 5 shows the two process areas included in CMMI-DEV level 5:

Table 5. CMMI-DEV level 5 process areas.

Process area	Category	Specific goals	Specific practices
Causal Analysis and Resolution (CAR)	Support	2	5
Organizational Performance Management (OPM)	Process Management	3	10

Gap analysis

This section will present our own mapping between Scrum and XP practices and the different CMMI levels 4 and 5 goals. For this purpose, we will assess the particular practices suggested against Scrum and XP practices, so as to identify those that will allow matching the specific goals and linking CMMI goals, if applicable, to the specific Web characteristics. Tables 6 to 9 represent the results of our Scrum/XP specific practices gap analysis per process area. They show that no practices either in Scrum or in XP cover these maturity levels goals.

Table 6. CMMI-level 4 gap analysis: OPP process area.

		CMMI level 4 Gap analysis – Organizational process performance					
		Scrum practice	Scrum coverage	XP practice	XP coverage	Specific goal coverage	Web characteristic
OPP-SG 1: Establish Performance Baselines and Models	SP 1.1: Establish Quality and Process Performance Objectives	None	--	None	--	--	None
	SP 1.2: Select Processes	None	--	None	--		None
	SP 1.3: Establish Process Performance Measures	None	--	None	--		None
	SP 1.4: Analyze Process Performance and Establish Process Performance Baselines	None	--	None	--		None
	SP 1.5: Establish Process Performance Models	None	--	None	--		None

Table 7. CMMI-level 4 gap analysis: QPM process area.

		CMMI level 4 Gap analysis – Quantitative project management					
		Scrum practice	Scrum coverage	XP practice	XP coverage	Specific goal coverage	Web characteristic
QPM-SG 1: Prepare for Quantitative Management	SP 1.1: Establish the Project's Objectives	None	--	None	--	--	None
	SP 1.2: Compose the Defined Process	None	--	None	--		None
	SP 1.3: Select Subprocesses and Attributes	None	--	None	--		None
	SP 1.4: Select Measures and Analytic Techniques	None	--	None	--		None
QPM-SG 2: Quantitatively Manage the Project	SP 2.1: Monitor the Performance of Selected Subprocesses	None	--	None	--	--	None
	SP 2.2 Manage Project Performance	None	--	None	--		None
	SP 2.3 Perform Root Cause Analysis	None	--	None	--		None

Table 8. CMMI-level 5 gap analysis: CAR process area.

		CMMI level 5 Gap analysis – Causal analysis and resolution					
		Scrum practice	Scrum coverage	XP practice	XP coverage	Specific goal coverage	Web characteristic
CAR-SG 1: Determine Causes of Selected Outcomes	SP 1.1: Select Outcomes for Analysis	None	--	None	--	--	Wb4, Wb5, Wb6
	SP 1.2: Analyze Causes	None	--	None	--		Wb4, Wb5, Wb6
CAR-SG 2: Address Causes of Selected Outcomes	SP 2.1: Implement Action Proposals	None	--	None	--	--	Wb4, Wb5, Wb6
	SP 2.2: Evaluate the Effect of Implemented Actions	None	--	None	--		Wb4, Wb5, Wb6
	SP 2.3: Record Causal Analysis Data	None	--	None	--		Wb4, Wb5, Wb6

Table 9. CMMI-level 5 gap analysis: OPM process area.

		CMMI level 5 Gap analysis – Organizational performance management					
		Scrum practice	**Scrum coverage**	**XP practice**	**XP coverage**	**Specific goal coverage**	**Web characteristic**
OPM-SG 1: Manage Business Performance	SP 1.1: Maintain Business Objectives	None	--	None	--	--	Wb5, Wb6
	SP 1.2: Analyze Process Performance Data	None	--	None	--		Wb5, Wb6
	SP 1.3 Identify Potential Areas for Improvement	None	--	None	--		Wb5, Wb6
OPM-SG 2: Select Improvements	SP 2.1: Elicit Suggested Improvements	None	--	None	--	--	Wb5, Wb6
	SP 2.2: Analyze Suggested Improvements	None	--	None	--		Wb5, Wb6
	SP 2.3: Validate Improvements	None	--	None	--		Wb5, Wb6
	SP 2.4: Select and Implement Improvements for Deployment	None	--	None	--		Wb5, Wb6
OPM-SG 3: Deploy Improvements	SP 3.1: Plan the Deployment	None	--	None	--	--	Wb5, Wb6
	SP 3.2: Manage the Deployment	None	--	None	--		Wb5, Wb6
	SP 3.3 Evaluate Improvement Effects	None	--	None	--		Wb5, Wb6

As a main conclusion of the specific practices gap analysis, we can state that neither Scrum nor XP can cover any of the maturity levels 4 and 5 practices. The two level 4 process areas focus on establishing baselines and measuring project and process performance, and the two analyzed methodologies do not propose any practices in this field. The two level 5 process areas aim at implementing quality and productivity, by avoiding the introduction of defects and managing organizational performance, as well as by analyzing aggregated project data. Besides, neither of the two studied methodologies suggests a practice in these fields.

We can also identify that level 4 process areas might not be related specifically to Web specific characteristics. On the contrary, we can relate level 5 ones to some of them. CAR can help to increase maintenance efficiency, as it will support systematic identification of root cause, together with reduction of time to market and delivery time (f.i. improved processes will reduce development and testing times). OPM can also assist in delivery time and adaptation to changes, as it is associated with improving organization performance, and Agile organizations performance highly depends on it.

Extending Scrum and XP to achieve CMMI-DEV levels 4 and 5 goals

As it has been stated, neither Scrum or XP nor a combined implementation of both could help to achieve CMMI maturity levels 4 and 5 goals for Web environment. In this subsection we will recommend some other Agile techniques and/or ad-hoc extensions to meet the proposed goals. Tables 10 and 11 show the recommended extensions:

Table 10. Proposed extensions to cover specific goals of CMMI maturity level 4.

			Proposed extensions
OPP	**OPP-SG 1: Establish Performance Baselines and Models**	SP 1.1: Establish Quality and Process Performance Objectives	**Ad-hoc extension to standard Scrum/XP implementation:** OPP deals with establishing organizational process performance objectives based on business objectives by means of defining measures, baselines and models. In this case, there is no specific Agile technique that suits the practice of this process area, so we propose the following ad-hoc extension: • Definition of performance baselines and Key Performance Indicators (KPIs) during the methodology rollout at organization level, based on Schwaber's techniques [34]. Some examples of this KPI could be: o Revenue per feature. o Produced business value. o Cycle time. o Defects identified after delivery. o Iteration cycle to review KPIs and its usefulness.
		SP 1.2: Select Processes	
		SP 1.3: Establish Process Performance Measures	
		SP 1.4: Analyze Process Performance and Establish Process Performance Baselines	
		SP 1.5: Establish Process Performance Models	
QPM	**QPM-SG 1: Prepare for**	SP 1.1: Establish the Project's Objectives	**Ad-hoc extension to standard Scrum/XP implementation:**
		SP 1.2: Compose the Defined Process	

	Quantitative Management	SP 1.3: Select Subprocesses and Attributes	The purpose of QPM is to quantitatively manage the project to achieve project's established quality and process performance objectives. We propose the following ad-hoc extensions to cover the goals of this process area: • During Sprint 0: o Adapt the process to achieve desired quality and performance objectives. o Select measures and techniques to apply to quantitative management. • During the rest of sprints: o Use Agile performance indicators proposed by Downey and Sutherland [11] to measure teams performance. o Use Agile EVM techniques [38, 41] to measure projects performance.
		SP 1.4: Select Measures and Analytic Techniques	
	QPM-SG 2: Quantitatively Manage the Project	SP 2.1: Monitor the Performance of Selected Subprocesses	
		SP 2.2 Manage Project Performance	
		SP 2.3 Perform Root Cause Analysis	

CMMI maturity level 4 focuses on the definition of baselines and models that will help to quantitatively manage the organization's projects. For OPP goals, we propose an ad-hoc extension, based on an initial definition of Agile KPIs together with the baselines, using concepts such as business value, cycle time or revenue per feature, and then establishing an organization iteration cycle to review them as well as their usefulness. To cover the specific goals of QPM, we recommend an ad-hoc extension focused on the use of a Sprint 0 to tailor the measures and processes for the specific project, with the aim to quantitatively manage it with elements like Agile performance indicators or Agile EVM.

Table 11. Proposed extensions to cover specific goals of CMMI maturity level 5.

			Proposed extensions
C A R	**CAR-SG 1: Determine Causes of Selected Outcomes**	SP 1.1: Select Outcomes for Analysis	**Lean Software Development:** CAR deals with identifying and addressing causes of outcomes. Most of the time these outcomes are defects or problems to correct, with the aim of avoiding them to be introduced in the product before its development. These concepts of identifying root causes and preventing the appearance of errors are quite common both in Goldratt's Theory of Constraints [39] and in the "Lean" manufacturing movement [17]. Tom and Mary Poppendieck translated them from the manufacturing world into software development environments by defining the Lean Software Development [29, 30, 31]. Coming from this methodology there are several techniques that can help addressing the five specific practices of this process area: • "*Build integrity in*" [31], including the tools of "perceived integrity", "conceptual integrity", "refactoring" and "testing". • "*Poka yoke*" processes, to make it difficult to introduce errors [30]. • "*Exposing problems*" to be aware of the current situation [29]. • "*Go to the workplace*" to experience in first hand the impact of the problems [29].
		SP 1.2: Analyze Causes	
	CAR-SG 2: Address Causes of Selected Outcomes	SP 2.1: Implement Action Proposals	
		SP 2.2: Evaluate the Effect of Implemented Actions	
		SP 2.3: Record Causal Analysis Data	
O P M	**OPM-SG 1: Manage Business Performance**	SP 1.1: Maintain Business Objectives	**Lean Software Development:** OPM allows organizations to iteratively manage organizational performance. OPM deals with project data analysis, business performance gap identification and improvements deployment to fill in these gaps. Again, the goals of this process area are linked to the Lean principles and should be managed using their approach: gather and analyze data, define a course of actions and evaluate the results, always with an iterative approach. The proposed technique for this process area is: • Continuous improvement process [18], to ensure that the process is periodically reviewed and problems are identified and addressed using Agile retrospectives techniques [9].
		SP 1.2: Analyze Process Performance Data	
		SP 1.3 Identify Potential Areas for Improvement	
	OPM-SG 2: Select Improvements	SP 2.1: Elicit Suggested Improvements	
		SP 2.2: Analyze Suggested Improvements	
		SP 2.3: Validate Improvements	
		SP 2.4: Select and Implement Improvements for Deployment	
	OPM-SG 3: Deploy Improvements	SP 3.1: Plan the Deployment	
		SP 3.2: Manage the Deployment	
		SP 3.3 Evaluate Improvement Effects	

CMMI maturity level 5 process areas is oriented towards identifying defects and problems early in the process, ideally before they are introduced, and towards business and organization-wide continuous improvement. As mentioned, these principles are also found in the Lean movement, from the manufacturing world. This movement includes elements and techniques like "continuous improvement", "poka yoke" (mistake-proof processes) or "stop and fix" that have been translated into software development by the Lean Software Development methodology [29, 30, 31]. Together with the Lean techniques, the use of Agile retrospectives in the Scrum/XP cycles will allow iteratively gathering data, identifying the most important problems, selecting the desired improvements and deploying them. The goals of both CAR and OPM process areas could be achieved with this approach, being the organization able to obtain CMMI level 5.

6. Conclusions and Future Work

The present work tries to provide some insight on the general question about the feasibility of providing an Agile approach to CMMI-DEV maturity levels 4 and 5. This general approach aims basically to:

1 Perform a gap analysis between Scrum/XP and CMMI-DEV levels 4 and 5 process areas, taking into account Web specifities.

2 Propose Agile techniques, either already existent or ad-hoc extensions, to fill in the identified gap.

As explained before, the generic research question was structured in three specific research questions that were linked to the goals described above. The first one was: *"What is the gap, if any, between Scrum and XP practices approaches and CMMI levels 4 and 5 goals?"* To get an answer, we conducted a detailed review of the existing literature, together with our own gap analysis at practice level for all CMMI-DEV maturity level goals. As a conclusion of these gap analyses, we highlight that neither Scrum nor XP can cover any CMMI-DEV levels 4 and 5 goals. At the same time the gap analysis was run, and by linking each of CMMI process areas to Web specific characteristics, we answered the second question: *"How CMMI levels 4 and 5 goals can be related at the same time to Web specificities and to Scrum/XP practices?"*. As a conclusion, we found that, although CMMI level 4 process areas might not be related to Web specificities, level 5 process areas might help, if implemented in an Agile way, to fit Web special needs.

Finally, once the gap was identified, we moved to the third research question*: "In case of a gap between Scrum/XP practices and CMMI levels 4 and 5 goals, are there any other Agile techniques that can fill in such a gap?"* To provide an answer, we identified a complete proposal with the intention to cover all specific CMMI-DEV levels 4 and 5 goals, by combining Scrum and XP practices with some other Agile techniques, such as Lean Software Development or Agile retrospectives, and also incorporating some "ad-hoc" modifications.

Combining this work with our previous ones [42, 43], we might have a complete proposal to achieve all CMMI maturity levels goals, suitable for Web environments. Currently, we are conducting a process based on an expert's judgement, with the aim to validate the model that will be explained in a future paper. As a future work, the presented analysis should be validated practically, either by means of a formal SCAMPI assessment or by self-assessment, in order to check the full coverage of CMMI goals in the real world. Another line of research should deal with the formalization of the proposed techniques and combine them into a coherent and well-defined framework, so as to offer organizations a mature Agile approach to progress through CMMI-DEV maturity levels. Integrating into this framework some of Web Engineering best practices, such as those proposed by NDT [12], will also increase the value of the proposal. Finally, the possibility of extending this research to development projects other than Web-related projects, can also be considered as a future line of work.

Acknowledgements

This research work has been supported by the MeGUS Project (TIN2013-46928-C3-3-R) of the Ministry of Economy and Competitiveness, Spain.

References

1. Ambler, S.W. "Lessons in Agility from Internet-based Development." IEEE Software, pp. 66-73. Mar-Apr, 2002.

2. Beck, K. et al. "Manifesto for Agile Software Development". 2001. From http://www.agilemanifesto.org. Last accessed 04-2016.
3. Beck, K.; Andres, C. 2004. "Extreme Programming Explained: Embrace Change, Second Edition". Boston: Addison-Wesley.
4. Begel, A.; Nagappan, N. 2007. "Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study. In proceedings of the 1st International Symposium on Empirical Software Engineering and Measurement (Madrid, Spain, September 21-27 2007) ESEM '07, IEEE.
5. Bougroun, Z.; Zeaaraoui, A.; Bouchentouf, T. 2014. "The projection of the specific practices of the third level of CMMI model in agile methods: Scrum, XP and Kanban". In proceedings of Information Science and Technology (CIST), 2014 Third IEEE International Colloquium in (pp. 174-179). IEEE.
6. Budzier, A.; Flyvbjerg, B. 2013. "Making sense of the impact and importance of outliers in project management through the use of power laws". In proceedings of IRNOP (International Research Network on Organizing by Projects), At Oslo, 11.
7. CMMI Product Team. "CMMI for Development, Version 1.3." Carnegie Mellon University. 2010. From http://www.sei.cmu.edu/reports/10tr033.pdf. Last accessed 04-2016.
8. Deshpande, Y.; Marugesan, S.; Ginige, A.; Hanse, S.; Schawabe, D.; Gaedke, M.; White, B. "Web Engineering". Journal of Web Engineering. Vol. 1 Nº 1, pp. 3-17.2002. Rinton Press.
9. Derby, E.; Larsen, D. "Agile Retrospectives. Making Good Teams Great". Dallas: The Pragmatic Bookshelf. 2006.
10. Díaz, J.; Garbajosa, J.; Calvo-Manzano, J.A. 2009. "Mapping CMMI Level 2 to Scrum Practices: An Experience Report". Jan. 2009. SPI 42 93-104.
11. Downey, S.; Sutherland, J. "Scrummetrics for hyperproductive teams: how they fly like fighter aircraft". In proceedings of the 45th Hawaii International Conference on System Science, 2012, Maui, Hawaii, USA, January 4–7 2012.
12. Escalona, M.J.; Aragón, G. 2008. "NDT: A Model-Driven Approach for Web requirements", IEEE Transactions on Software Engineering, 34(3). 2008. 370-390.
13. Escalona, M. J.; Mejías, M.; Torres, J. 2004. "Developing systems with NDT & NDT-Tool". In 13th International conference on information systems development: methods and tools, theory and practice, Vilna, Lithuania (pp. 149-59).
14. Goldenson, D. R.; Gibson, D. L. ; Ferguson, R. L. "Why Make the Switch? Evidence about the Benefits of CMMI". From http://www.sei.cmu.edu/library/assets/evidence.pdf. Last accessed 04-2016.
15. Goldratt, E. "The Goal". The North River Press Publishing Corporation, Third Revision, 2004.
16. Joshi, J. B.; Aref, W. G.; Ghafoor, A.; Spafford, E. H. 2001. "Security Models for Web-based Applications". Communications of the ACM, 44(2), 38-44.
17. Kennedy, M. "Product Development for the Lean Enterprise: Why Toyota's System Is Four Times More Productive and How You Can Implement It". The Oaklea Press, April 2003.
18. Kniberg, H. "Lean from the Trenches: Managing Large-Scale Projects with Kanban". Dallas: The Pragmatic Bookshelf. 2012.
19. Koch, N.; Knapp, A.; Zhang, G.; Baumeister, H. 2008. "UML-Based Web Engineering: An Approach Based on Standards". Web Engineering: Modelling and Implementing Web Applications, Springer, pp. 157-191. 2008.
20. Lukasiewicz, K.; Miler, J. 2012. "Improving Agility and Discipline of Software Development with the Scrum and CMMI," Software, IET, vol.6, no.5, pp.416, 422, October 2012.
21. Marcal, A.S.C.; de Freitas, B.C.C.; Furtado Soares, F.S.; Belchior, A.D. 2008. "Blending Scrum Practices and CMMI Project Management Process Areas". 2008. ISSE 4. 17-29.

22. Mendes, E.; Mosley, N. “Web Cost Estimation: An Introduction”. Web Engineering: Principles and Techniques. IGI Global, pp 182-202. 2005.
23. Model Driven Web Engineering Workshop. Satellite Workshop of ICWE’2012 Conference. From http://mdwe2012.pst.ifi.lmu.de. Last accessed 04-2016.
24. Murugesan, S.; Deshpande, Y.; Hansen, S.; Ginige, A. 2001. “Web engineering: A new discipline for development of web-based systems”. In Web Engineering (pp. 3-13). Springer Berlin Heidelberg.
25. Object Management Group. 2014. “IFML: The Interaction Flow Modeling Language”. From http://www.ifml.org/?page_id=99. Last accessed 03-2015.
26. Paulk, M. C. 2001. “Extreme programming from a CMM perspective”. Software, IEEE, 18(6), 19-26.
27. Pikkarainen, M. et al. 2008. “The Impact of Agile Practices on Communication in Software Development”. Empirical Software Engineering, Springer, pp. 303-337. May. 2008.
28. Pressman, R.S. “What a Tangled Web We Weave”. IEEE Software, pp 18-21. Jan.-Feb. 2000.
29. Poppendieck, M. et al. “Leading Lean Software Development”. Boston:Addison-Wesley, 2009.
30. Poppendieck, M. et al. “Implementing Lean Software Development”. Boston:Addison-Wesley, 2007.
31. Poppendieck, M.; Poppendieck, T. 2003. “Lean Software Development. An Agile Toolkit”. Boston: Addison-Wesley.
32. Ran, H. et al. “Agile Web Development with Web Framework”. Proceedings of the 4th Int. Conf. on Wireless Communications, Networking and Mobile Computing. (Dalian, China, October 12 – 17 2008) WiCOM '08. IEEE.
33. Reifer, D.J. 2000. “Web Development: Estimating quick-to-market software. IEEE Software, Nov - Dec, (pp. 57-64).
34. Schwaber, K. “The Enterprise and Scrum”. Redmond: Microsoft Press. 2007
35. Selleri Silva, F.; Santana Furtado Soares, F.; Lima Peres, A.; Monteiro de Azevedo, I; Vasconcelos, A.; Kenji Kamei, F.; Romero de Lemos Meira, S. 2015. “Using CMMI together with agile software development: A systematic review”. Information and Software Technology, Volume 58, February 2015, Pages 20-43.
36. Serrador, P.; Pinto, J.K. “Does Agile work? — A quantitative analysis of agile project success”. International Journal of Project Management, Volume 33, Issue 5, July 2015, Pages 1040-1051.
37. Staples, M.; Niazi, M.; Jeffery, R.; Abrahams, A.; Byatt, P.; Murphy R. 2007. “An exploratory study of why organizations do not adopt CMMI”. J. Syst. Softw. 80, 6 (June 2007), 883-895.
38. Sulaiman T.; Barton, B.; Blackburn, T. “AgileEVM – Earned Value Management in Scrum Projects”. In proceedings of the Agile Conference, 2006, Minneapolis, Minnesota, 23–28 July 2006.
39. Sutherland, J.; Schwaber, K. 2011. “The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game”. From http://www.scrum.org/Scrum-Guides. Last accessed 01-2016.
40. Torrecilla Salinas, C.J.; Sedeño, J.; Escalona, M.J.; Mejías, M. 2015. “Agile, Web Engineering and Capability Maturity Model Integration: A systematic literature review”. InfSoftwTechnol 71 (2016) 92–107.
41. Torrecilla Salinas, C.J.; Sedeño, J.; Escalona, M.J.; Mejías, M. 2015. “Estimating, planning and managing Agile Web development projects under a value-based perspective”. InfSoftwTechnol 61 (2015) 124–144.
42. Torrecilla Salinas, C.J.; Sedeño, J.; Escalona, M.J.; Mejías, M. 2014. “Mapping Agile Practices to CMMI-DEV Level 3 in Web Development Environments”. In Information Systems Development: Transforming Organisations and Society through Information Systems (ISD2014 Proceedings). Varaždin, Croatia.

43. Torrecilla Salinas, C.J.; Escalona, M. J.; Mejías, M. 2012. “A Scrum-based Approach to CMMI Maturity Level 2 in Web Development Environments”. In proceeding of the International Conference on Information Integration and Web-based Applications & Services 2012 (Bali, Indonesia December 3-5 2012). iiWAS, 12. ACM.
44. VersionOne. “9th Annual State of Agile Survey”. From http://www.versionone.com/pdf/state-of-agile-development-survey-ninth.pdf. Last accessed 04-2016.