

Mapping Agile Practices to CMMI-DEV Level 3 in Web Development Environments*

C.J. Torrecilla-Salinas

carlos.torrecilla@iwt2.org

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

M. Mejías

risoto@us.es

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

Abstract

Agile approaches formally appeared ten years ago and nowadays are a valid alternative for organizations developing software. Agile methodologies are especially interesting to those developing Web environments applications, as they can fit properly the special characteristics of this type of developments. In addition, maturity models like CMMI-DEV (Capability Maturity Model Integration for Development) focus on assessing the maturity level of organizations developing or acquiring software. These models are well established and can increment quality of development processes to enhance costumers' satisfaction. CMMI-DEV level 3 provides a good compromise on maturity gained and effort needed. The feasibility of getting it through a combination of Agile methods can be very useful to organizations developing systems in Web environments, as they can keep the adaptability of Agile together with a more mature development process. This paper proposes a set of Agile methods so as to reach all CMMI-DEV maturity level 3 generic and specific goals. Based on this analysis, the paper proposes further research lines.

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

1. Introduction

CMMI-DEV (Capability Maturity Model Integration for Development) is part of CMMI (Capability Maturity Model) family. This maturity model provides a comparative framework to assess the maturity level that organizations can reach when developing or acquiring software solutions [10]. It is assumed that the fact of achieving the different CMMI maturity levels is related to product quality improvements [21] and more than 5.000 companies are using CMMI all over the world [9]. Besides, Web-based developments have special characteristics that differentiate them from classical development projects, such as a complex navigational structure; critical interface requirements, (such as unknown users or availability, among others); security aspects; increase on maintenance efficiency, avoiding downtimes; delivery as soon as possible; reduction of "time-to-market" and adaptation to quick-changing requirements [5, 15, 37]. Some of these characteristics, for instance, reducing "time-to-market" or quickly adaptation to undefined requirements, are becoming more and more important in Web projects [36]. As it is known, one of the principles of Agile approaches is to embrace changes [5], thus

* The views presented on this paper are those of their authors, and do not necessarily reflect those of their employers

Agile approaches offer a suitable framework for the exposed Web development characteristics [37], such as quick response to changes, adaptability and reduction of development time [8, 9]. In addition, classical approaches regarding up-front requirements gathering demands a stable environment. This is not the case of Web projects, where requirements are constantly changing. The incremental and iterative way of processing requirements provided by Agile [17] may better fit this particular case. In the last years, a growing trend can be observed in the use of Agile, including major companies like Microsoft, Amazon or Yahoo. This trend is also observed in Web environments [1]. However, the more Web applications are becoming popular, the more their quality requirements are increasing. As mentioned, higher levels of CMMI-DEV maturity model are associated with quality improvements. Thereby, the usage of Agile methods to achieve the proposed goals of CMMI-DEV maturity levels could offer organizations developing Web environments the possibility of combining quality and maturity levels with the ability to respond to changes, even when sometimes both approaches, Agile and CMMI, have been regarded as opposite; as both include valid principles for software development that are not necessarily incompatible [20]. In particular, CMMI-DEV level 3 can present a good compromise between CMMI formalism and Agile adaptability [13, 28]. Based on the foregoing, this work identifies the following objectives: map a set of Agile practices with CMMI-DEV level 3 goals; take out the relevant conclusions and identify the future lines of research. For this purpose, this paper is organized into the following sections: after this introduction, Section 2 offers an overview of Agile and CMMI and Section 3 summarizes the related work. Section 4 presents a detailed mapping between the identified Agile practices to CMMI-DEV level 3 specific and generic goals. Finally, Section 5 states the main conclusions and further lines of research.

2. Overview of Agile Methodologies and CMMI-DEV

2.1. Agile Methodologies

Agile is a label which groups a set of different methodologies and techniques that appeared in software development during the last decade of the 20th century, as an evolution of the previously existent iterative and incremental approaches. The main goals of these practices were, firstly, to ensure that valuable results were delivered to customers and users as soon as possible, and lastly, to allow development organizations to adapt easily their products to users' changing requirements. All of these approaches shared the values and principles stated in the "Agile Manifesto" [5]: collaboration between development team and business, quick response to changes even in late phases of development, short feedback cycles, early delivery of value or focus on technical excellence. Some of the most popular Agile approaches are: Scrum [42], eXtreme Programming (XP) [6] or Lean Software Development [35]. However, Scrum is, by far, the most common Agile method, which is used either alone or combined with other Agile techniques [34].

2.2. CMMI-DEV

The Capability Maturity Model Integration is an approach to process improvements for organizations to develop effective processes [10]. CMMI includes CMMI-DEV, the maturity model for organizations building or acquiring software, whose current version is 1.3 [10]. This updated version includes a set of practices in twenty-two process areas, and it is structured into levels in order to help organizations better their development processes. The model recommends two representations, named Continuous and Staged respectively, that depict different improvement paths for organizations. Our work will be based on the Staged representation, which focuses on the global maturity level of an organization considering it a way of characterizing its performance. In the Staged representation the organization improves a subset of processes in each of the maturity levels, preparing them to the next one. Five maturity levels are defined in this type of representation: Initial, Managed, Defined,

Quantitative Managed and Optimized, and Table 1 shows the set of CMMI-DEV level 3 process areas in the Staged representation:

Table 1. CMMI-DEV maturity level 3 process areas.

| Process Area | Category | Process Area | Category |
|---|--------------|-------------------------------|--------------|
| Decision Analysis and Resolution (DAR) | Support | Requirements Development (RD) | Engineering |
| Integrated Project Management (IPM) | Project Mgmt | Risk Management (RSKM) | Project Mgmt |
| Organizational Process Definition (OPD) | Process Mgmt | Technical Solution (TS) | Engineering |
| Organizational Process Focus (OPF) | Process Mgmt | Validation (VAL) | Engineering |
| Organizational Training (OT) | Process Mgmt | Verification (VER) | Engineering |
| Product Integration (PI) | Engineering | | |

3. Related Works

This section presents in detail the existing related works on this topic, considering any approach, coming from Agile world trying to incorporate values from CMMI, or vice versa. We have paid special attention to studies focusing on Web environments. As a result, we have found approaches like the ones by Morales *et al.* [32], Jakobsen *et al.* [24, 25], Sutherland *et al.* [20] and Maller *et al.* [29], describing how different organizations with a strong CMMI culture incorporate Agile practices (like Scrum, XP, Lean or TDD) in their development processes. These studies have some common patterns: They have the form of case studies, analyze organizations already being assessed in higher levels of CMMI, present a slightly detailed high-level mapping between Agile practices and CMMI goals and all focus on general development processes, without including Web development particularities. The main gap of such literature is that authors explain in general terms, how an already CMMI-assessed organization incorporates Agile practices. Nevertheless, they neither describe how an organization can progress through CMMI levels using Agile practices nor how to map Agile practices with CMMI goals (even if some of them cope with the latter briefly). We have also found works where the process of Agile organizations starting with formal assessments on different CMMI maturity levels is described, like these by Cohan *et al.* [11], Baker [3, 4], Garzas *et al.* [19] and Bon *et al.* [8]. These papers present some case studies dealing with how companies using Scrum or XP successfully went through a formal CMMI assessment. In Garzas' paper, only CMMI level 2 is assessed against Scrum, without analyzing maturity level 3 goals. The remaining papers are centered on the preparation of the assessment process and not on describing the mapping between the different practices and goals, which either is not presented or it is done at a very high-level. These papers also point to general software development processes, without including Web projects peculiarities. In contrast, Miller *et al.* [31] present a case study regarding how a company started with an Agile software development implementation based on Scrum methodology and a formal assessment process at CMMI maturity level 2 at the same time. This approach analyzes the problem from the general development perspective by presenting the mapping in a non-detailed way, only studying maturity level 2 goals. As the progress on CMMI-DEV level is linked to quality improvements [21], our work extends the analysis to maturity level 3.

In addition, a set of theoretical works has been gathered, like those by Lukasiewicz *et al.* [28], Zang *et al.* [44], Marçal *et al.* [30], Omran *et al.* [33] and Díaz *et al.* [16]. These studies introduce a mapping between a certain set of Agile practices (mainly Scrum or a variation of Scrum) and the goals of a certain CMMI maturity level. Lukasiewicz's work introduces a mapping between Scrum and some process areas of CMMI levels 2 and 3. Then, a variation of Scrum is proposed to fill in the gaps. Nevertheless, it focuses on generic developments, without taking into account Web specificities and without covering all process areas. Marçal's work analyzes the mapping between Scrum and CMMI project management process areas of maturity levels 2, 3 and 4, but there is no proposal on how to fill in the identified gaps. Díaz maps Scrum and CMMI level 2 also from a theoretical point of view, but only covering some process areas, without considering Web specificities, neither. In contrast, we propose a mapping for the full

set of process areas of CMMI-DEV level 3, not only using Scrum, but also some other Agile practices and methods that can cope with Web projects particularities.

As a conclusion, we can state that there is no work proposing a full mapping between a set of Agile practices and methods and all CMMI-DEV maturity levels 2 and 3 process areas that considers Web projects particularities, with the exception of our previous work [43]. It analyzes the mapping between Scrum and the full set of CMMI level 2 process areas for Web development environments, as well as recommends a set of Agile practices to fill the identified gaps, by proposing an Agile framework to reach all level 2 goals. This is the only work focusing on Web projects. Therefore we can consider it to be the starting point for this study. The conclusions of this previous work allows extending our study to all CMMI-DEV level 3 process areas by intending to map a different set of Agile practices (not limited to Scrum practices) to all specific and generic goals of all CMMI maturity level 3 process areas, taking into account Web development projects characteristics.

4. Mapping Proposal

As previously mentioned, our work [43] recommended a set of Agile practices covering all specific and generic goals of CMMI-DEV level 2 process areas. In this section we offer a mapping between a set of Agile practices and CMMI-DEV maturity level 3 specific and generic goals for Web development projects. This mapping proposal has been designed by analyzing the description of each one of the process areas goals and its proposed practices, to later matching them with the description of the Agile practices and techniques provided in the literature.

4.1. Specific Goals

Table 2 summarizes the proposal for CMMI-DEV maturity level 3 specific goals:

Table 2. Results for CMMI-DEV maturity level 3 generic goals.

| P. A. | Goal | Proposed Agile techniques |
|-------|--|---|
| DAR | SG 1 Evaluate Alternatives | DAR supports the analysis of possible decision-making processes using a formal evaluation procedure. At a project level, the process of identifying, evaluating and selecting alternatives can be performed by means of the Highsmith's techniques in the "Envisioning" phase of his Agile Project Management approach [22]. This phase will allow assessing the feasibility of projects systematically, identifying their scope and budget, prioritizing them in terms of value and planning them at a high level in liaison with the project's stakeholders. The set of practices included in this phase would cover the six specific practices of this specific goal. |
| IPM | SG 1 Use the Project's Defined Process | The goal of IPM is to manage the project involving stakeholders in it. Scrum [42] is suggested as the basis to achieve the goals of this process area, covering at least three SG2 specific practices. Using Schwaber's approach to implement Scrum at enterprise level [39], which would allow institutionalizing Scrum as an organization standard process, is proposed in order to cope with seven SG1 specific practices. |
| | SG 2 Coordinate and Collaborate with Relevant Stakeholders | |
| OPD | SG 1 Establish Organizational Process Assets | OPD aims to establish and maintain a usable set of organizational process assets, work environment standards, and rules and guidelines for teams. Using Schwaber's process [39] to implement Scrum at enterprise level can be useful to cover the first three specific practices, although there are no particular Agile practices to establish processes at organization level. Besides, at this level, the establishment of ground rules for the teams and "definitions of done" [42], as prescribed by Scrum, can be useful so as to establish the organization work standards, covering 6 and 7 specific practices. The extension to Scrum proposed by our previous work [43] can be also applied to the organization measurement repository, allowing tackling 4 and 5 specific practices. |
| O P F | SG 1 Determine Process Improvement Opportunities | |

| | | |
|------|---|--|
| | SG 2 Plan and Implement Process Actions | OPF focuses on planning, implementing and deploying organizational process improvements. The Agile field providing support to the goals of this process area can be Lean Software Development, including: - Agile retrospectives at project and enterprise level [14, 35] for SG1 and SG2 specific practices. - Process improvement workshops [26] for SG 2 and SG3 specific practices. |
| | SG 3 Deploy Organizational Process Assets and Incorporate Experiences | |
| OT | SG 1 Establish an Organizational Training Capability | OT deals with developing the team's skills and knowledge. An organizational training plan should be implemented with "Amplify learning" practice of Lean Software Development [35] to achieve this goal. Reducing feedback and learning cycles and working on iterations will suit Web development characteristics properly and allow covering the seven specific practices of the two goals |
| | SG 2 Provide Training | |
| PI | SG 1 Prepare for Product Integration | PI's goal concerns to ensemble the final product. Web development demands reducing "time-to-market" as much as possible. Agile practices below can be implemented in order to achieve this goal together with that of this process area: - Continuous Integration [6], which will cover 1 and 2 specific goals practices. - Continuous Delivery [23], which will also cover 3 specific goal practices. |
| | SG 2 Ensure Interface Compatibility | |
| | SG 3 Assemble Product Components and Deliver the Product | |
| RD | SG 1 Develop Customer Requirements | RD is about eliciting and analyzing requirements. As requirements in Web environments are not often clear at the beginning of the project, the idea is to include user stories [12] combined with other techniques, such as Personas [38] and Storyboards [38], as elements of Scrum Product Backlog [42] that will evolve through the project. This approach will cover specific SG1 and SG2 practices. The use of Scrum framework to elicit, define, build and validate requirements in projects Sprints will guarantee that the rest of the specific practices will be covered. |
| | SG 2 Develop Product Requirements | |
| | SG 3 Analyze and Validate Requirements | |
| RSKM | SG 1 Prepare for Risk Management | RSKM has the goal of identifying potential problems and mitigate their adverse consequences. Even though Risk Management is not a field deeply developed in Agile, the following techniques can be used along with Scrum process to manage Web projects risks: - Agile Risk Management [27], to cover SG 1 specific practices. - Risk Burn-down charts [13], for SG2 and SG3 specific practices. |
| | SG 2 Identify and Analyze Risks | |
| | SG 3 Mitigate Risks | |
| TS | SG 1 Select Product Component Solutions | TS copes with selecting, designing and implementing solutions to requirements. As Web developments are characterized by short feedback cycles and fuzzy requirements, the use of the following practices will fit both, the goal of this process area and Web projects needs: - Spike Solutions [40] and Exploratory testing [40] for SG 1 practices. - Simple Design [40] and Incremental Design and Architecture [40], for SG2 and SG3 practices. |
| | SG 2 Develop the Design | |
| | SG 3 Implement the Product Design | |
| VAL | SG 1 Prepare for Validation | VAR deals with ensuring that the team builds "the right product". As mentioned, unknown and changing requirements characterize Web developments. Thus, in this case, Agile test practices might be quite useful. The proposed practices to achieve the process area goals are ATDD (Acceptance Test Driven Development) or "Specification by example" [2], which will cover all SG1 and SG2 practices. |
| | SG 2 Validate Product or Product Components | |
| VER | SG 1 Prepare for Verification | VER ensures that the team builds "the product the right way". There are several testing techniques that help achieve the goals of this process area, most of them coming from XP: - Continuous integration [6] for SG1 practices. - Pair programming [6] for SG2 practices. - Test-Driven Development [7] for SG1 and SG3 practices. |
| | SG 2 Perform Peer Reviews | |
| | SG 3 Verify Selected Work Products | |

4.2. Generic Goals

Table 3 presents the proposed set of Agile methods in relation to CMMI level 3 generic goals.

Table 3. Results for CMMI-DEV maturity level 3 generic goals.

| Generic goal | Comments |
|----------------------------|---|
| GG1 Achieve Specific Goals | The aforementioned set of Agile practices will ensure meeting this goal by achieving the specific goals of each process area. |

| | |
|--|---|
| GG2 Institutionalize a Managed Process | This goal is linked to setting a management process for the defined development process. Establishing practices at organization level by means of the techniques proposed for IPM and OPD process areas will enable completing this goal. |
| GG3 Institutionalize a Defined Process | This objective is linked to establishing a process at organization level, which can be adapted to each particular project. Agile practices proposed for IPM and OPD process areas will also be used to achieve this generic goal. |

5. Conclusions and Future Work

To conclude, it must be stated that a combination of different Agile techniques could be used to achieve all CMMI-DEV maturity level 3 generic and specific goals in a Web development environment. In this case, the approach is based on using Scrum framework in order to manage and guide the project. Scrum process provides an iterative and incremental framework to build products, although it should be implemented at enterprise level, in order to allow institutionalizing the process and build an enterprise-wide assets repository, which is tailored-made for each individual project. This set of practices covers mainly IPM and RD process areas. The process proposed to deploy Scrum at enterprise level can also be useful for OPD process area. A set of technical practices together with Scrum must be implemented to achieve engineering process areas goals. These practices come from XP and early testing worlds, such as continuous integration and delivery, pair programming, incremental design and spikes, specification by example and Test Driven Development, among others. Such practices cover PI, TS, VAL and VER. On the top of Scrum and along with technical practices, some other Agile methods help organizations reach CMMI-DEV maturity level 3: Continuous improvement and organizational learning coming from the Lean Software Development context to cover OPF and OT goals; Agile Risk Management approach and its set of tools to cover RSKM and Agile Project Management envisioning phase, in order to support the decision making process in an Agile way to cover DAR goals.

As it has been highlighted, this proposal is fully compatible with our proposal issued for CMMI-DEV level 2 in Web environments. Thus, the implementation of this set of practices will make an organization progress step by step through CMMI-DEV model, increasing its process maturity, but keeping its Agility.

A future line of work could consist in formalizing and integrating the proposed techniques in a consistent framework, as well as keeping the identification of Agile practices, methodologies and techniques that together can allow organizations producing Web developments to achieve the higher levels of CMMI-DEV model goals (4 and 5). Merging this set of practices with the already identified set for levels 2 and 3 will allow the definition of this consistent framework that will help in the process of institutionalizing Agile practices for Web development in a continuous improvement environment assessed by a widespread model like CMMI-DEV. As NDT [18] is a well-established Web development methodology, compatible with Agile lifecycles, we have realized that proposing one alternative, named Agile-NDT, will enable reaching the aforementioned goals. Finally, evaluating the model in a real-life implementation through a formal assessment or a self-assessment will be required in order to validate the proposals included in our work.

Acknowledgements

This research has been supported by the MeGUS project (TIN2013-46928-C3-3-R) of the Ministerio de Ciencia e Innovación, the Tempros project (TIN2010-20057-C03-02), and by the NDTQ-Framework project (TIC-5789) of the Junta de Andalucía, Spain.

References

1. Ambler, S.W.: Lessons in Agility from Internet-based Development. IEEE Software 19(2), 66-73 (2002)

2. Adzik, G.: *Specification by Example: How Successful Teams Deliver the Right Software*. Manning Publications, Shelter Island (2011)
3. Baker, S.W.: *Formalizing Agility: An Agile Organization's Journey Toward CMMI Accreditation*. Proceedings of Agile Conference, pp. 185-192. IEEE, Denver (2005)
4. Baker, S.W.: *Formalizing Agility, Part 2: How an Agile Organization Embraced the CMMI*. Jul. 2006. Proceedings of Agile Conference. IEEE, Minneapolis (2006)
5. Beck, K., Beedle, M., Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, K., Sutherland, J., Thomas, D.: *Manifesto for Agile Software Development* (2001), <http://www.agilemanifesto.org>, Accessed February 2014
6. Beck, K., Andres, C.: *Extreme Programming Explained: Embrace Change*, Second Edition. Addison-Wesley, Boston (2004)
7. Beck, K.: *Test Driven Development: By Example*. Addison-Wesley, Boston (2008)
8. Bos, E., Vriens, C.: *An Agile CMM*. In: Zannier, C., Erdogmus, H., Lindstrom, L. (eds) 4th Conference on Extreme Programming and Agile Methods. LNCS, vol. 3134, pp. 129-138. Springer, Heidelberg (2004)
9. CMMI Institute. *Published Appraisal Results*. Frok <https://sas.cmmiinstitute.com/pars/>, Accessed June 2014
10. CMMI Product Team: *CMMI for Development, Version 1.3*. Carnegie Mellon University (2010), <http://www.sei.cmu.edu/reports/10tr033.pdf>, Accessed February 2014.
11. Cohan, S., Glazer, H.: *An Agile Development Team's Quest for CMMI Maturity Level 5*. Proceedings of Agile Conference, pp. 201-206. IEEE, Chicago (2009)
12. Cohn, M.: *User Stories Applied: For Agile Software Development*. Addison-Wesley, Boston (2004)
13. Cohn, M.: *Managing Risk on Agile Projects with the Risk Burndown Chart*, <http://www.mountangoatsoftware.com/blog/managing-risk-on-agile-projects-with-the-risk-burndown-chart>, Accessed February 2014
14. Derby, E., Larsen, D.: *Agile Retrospectives. Making Good Teams Great*. The Pragmatic Bookshelf, Dallas (2006)
15. Deshpande, Y., Murugesan, S., Ginige, A., Hansen, S., Schwabe, D., Gaedke, M., White, B.: *Web Engineering*. Journal of Web Engineering 1(1), 3-17 (2002)
16. Díaz, J., Garbajosa, J., Calvo-Manzano, J.A.: *Mapping CMMI Level 2 to Scrum Practices: An Experience Report*. In: O'Connor, R.V. et al. (eds.) 16th European Conference on Software Process Improvement. Communications in Computer and Information Science, vol. 42, pp. 93-104. Springer, Heidelberg (2009)
17. Dybå, T., Dingsøy, T.: *Empirical Studies of Agile Software Development: A Systematic Review*. Information and Software Technology 50(9-10), 833-859 (2008)
18. Escalona, M.J., Aragón, G.: *NDT: A Model-Driven Approach for Web requirements*, IEEE Transactions on Software Engineering 34(3), 370-390 (2008)
19. Garzás, J., Paulk, M.C.: *A case study of software process improvement with CMMI-DEV and Scrum in Spanish companies*. Journal of Software: Evolution and Process 25(12), 1325-1333 (2013)
20. Glazer, H., Dalton, J., Anderson, D., Konrad, M.D., Shrum, S.: *CMMI or Agile: Why Not Embrace Both!* Carnegie Mellon University, Software Engineering Institute (2008), <http://www.sei.cmu.edu/reports/08tn003.pdf>, Accessed February 2014
21. Goldenson, D., Gibson, D.: *Demonstrating the Impact and Benefits of CMMI: An Update and Preliminary Results*. Carnegie Mellon University, Software Engineering Institute (2003), <http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=6365>, Accessed February 2014
22. Highsmith, J.: *Agile Project Management: Creating Innovative Products*, Second Edition. Addison-Wesley, New York (2009)
23. Humble, J., Farley, D.: *Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation*. Addison-Wesley, Boston (2010)

24. Jakobsen, C. R., Johnson, K.A.: Mature Agile With a Twist of CMMI. Proceedings of Agile Conference, pp. 212-217. IEEE, Toronto (2008)
25. Jakobsen, C.R., Sutherland, J.: Scrum and CMMI Going from Good to Great. Proceedings of Agile Conference, pp. 333-337. IEEE, Chicago (2009)
26. Kniberg, H.: Lean from the Trenches: Managing Large-Scale Projects with Kanban. The Pragmatic Bookshelf, Dallas (2012)
27. Lant, M.: Five Simple Steps to Agile Risk Management, <http://michaellant.com/2010/06/04/five-simple-steps-to-agile-risk-management/>. Accessed February 2014
28. Lukaszewicz, K., Miler, J.: Improving Agility and Discipline of Software Development with the Scrum and CMMI. IET Software 6(5), 416-422 (2012)
29. Maller, P., Ochoa, C., Silva, J.: Lightening the Software Production Process in a CMM Level 5 Framework. IEEE Latin America Transactions 3(1), 15-22 (2005)
30. Marçal, A.S.C., De Freitas, B.C.C., Furtado Soares, F.S., Furtado, M.E.S., Maciel, T.M., Belchior, A.D.: Innovations in Systems and Software Engineering 4, 17-29 (2008)
31. Miller, J.R., Haddad, H.M.: Challenges Faced While Simultaneously Implementing CMMI and Scrum: A Case Study in the Tax Preparation Software Industry. Proceedings of the 9th Information Technology: New Generations, pp. 314-318. IEEE, Las Vegas (2012)
32. Morales Trujillo, M., Oktaba, H., Pino, F.J., Orozco, M.J.: Applying Agile and Lean Practices in a Software Development Project into a CMMI Organization. In: Caivano, D. et al. (eds.) 12th International Conference on Product-Focused Software Process Improvement. LNCS, vol, 6759, pp. 17-29. Springer, Heidelberg (2011)
33. Omran, A.: AGILE CMMI from SMEs Perspective. Proceedings of the 3rd Int. Conf. on Information and Communication Technologies: From Theory to Applications, pp. 1-8. IEEE, Damascus (2008)
34. Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., Still, J.: The Impact of Agile Practices on Communication in Software Development. Empirical Software Engineering 13(3), 303-337 (2008)
35. Poppendieck, M., Poppendieck, T.: Lean Software Development. An Agile Toolkit. Addison-Wesley, Boston (2003)
36. Pressman, R.S.: What a Tangled Web We Weave. IEEE Software 17(1), 18-21 (2000)
37. Ran, H., Wang, Z., Hu, J., Xu, J., Xie, J.: Agile Web Development with Web Framework. Proceedings of 4th Int. Conf. on Wireless Communications, Networking and Mobile Computing, pp. 1-4. IEEE, Dalian (2008)
38. Rasmuson, J.: The Agile Samurai. How Agile Masters Deliver Great Software. The Pragmatic Bookshelf, Dallas (2011)
39. Schwaber, K.: The Enterprise and Scrum. Microsoft Press, Redmond (2007)
40. Shore, J., Warden, S.: The Art of Agile Development. O' Reilly (2007)
41. Sutherland, J., Jakobsen, C.R., Johnson, K.: Scrum and CMMI Level 5: The Magic Potion for Code Warriors. Proceedings of Agile Conference, pp. 272-278. IEEE, Washington (2007)
42. Sutherland, J., Schwaber, K.: The Scrum Guide: The Definitive Guide to Scrum (2011), <http://www.scrum.org/Scrum-Guides>, Accessed February 2014
43. Torrecilla-Salinas, C.J., Escalona, M.J., Mejias, M.: A Scrum-based Approach to CMMI Maturity Level 2 in Web Development Environments. Proceedings of the 14th International Conference on Information Integration and Web-based Applications & Services, pp. 282-285. ACM, Bali (2012)
44. Zhang, L., Dan, S.: Research on Combining Scrum with CMMI in Small and Medium Organizations. Proceedings of 2012 Int. Conf. on Computer Science and Electronics Engineering, pp. 554-557. IEEE, Hangzhou (2012)