Agile Methods and Cyber-Physical Systems Development—A Review with Preliminary Analysis

Muhammad Ovais Ahmad(✉)

Department of Mathematics and Computer Science, Karlstad University, Karlstad, Sweden
Ovais.ahmad@kau.se

Abstract. The software companies are using Agile methods and practices to tackle challenges in the rapidly changing environments and increasingly complex software systems. However, companies developing cyber physical systems (CPS) are still infancy in the use of Agile methods and hesitate to adopt. This systematic literature review was conducted in order to analyze the current trends of Agile methods use for CPS development. The search strategy resulted in 101 papers, of which 15 were identified as primary studies relevant to our research. The results show growing trend of Agile processes and Scrum is widely used reported for CPS development. The primary studies also exhibits a growing interest in teaching Agile in embedded systems, CPS and other engineering degree programs. The reported challenges included synchronization of software and hardware development, software and hardware developers use different vocabulary, lack of visibility and track of software releases and project progress. Additionally, lesson learned were extracted from the primary studies for guiding the practitioners interested in adopting Agile for CPS development.

Keywords: Agile · Scrum · Cyber physical systems · Software development

1 Introduction

In 2011, at Hannover fair, the term Industry 4.0 presented as fourth industrial revolution driven by the Internet [1]. The background on the term industry 4.0 came from the German government initiative to promote manufacturing industry using latest information and communication technologies. In general, there are forces which brings industrial revolution such as constant pressure to improve software, product and service quality, stay competitive in the market, enhance safety, sustainability and more important to remain profitable etc. To cope with these forces and become more agile the companies in every sector paying more attention to take advantages of technology and digitalization. The industry 4.0 focuses on interconnectivity, automation, machine learning, smart digital technology and real-time data. According to Hermann et al. [2] there are four key component of Industry 4.0: Cyber-physical systems (CPS), Internet of things (IoT), Internet of services (IoS), Smart factory. This digital transformation bring the traditional engineering closer to software engineering and create new interesting technical challenges and opportunities.

Y. Tian et al. (Eds.): ICBDS 2019, CCIS 1210, pp. 274–285, 2020.
https://doi.org/10.1007/978-981-15-7530-3_20

CPS is one of the important and fastest growing area in today ICT development paradigm. "*Cyber-physical systems (CPS) are physical and engineered systems whose operations are monitored, coordinated, controlled and integrated by a computing and communication core*" [8]. According to Leitao et al. [10] CPS extends the concept of embedded systems: "*in embedded systems the focus is on computational elements hosted in stand-alone devices, while CPS is designed as a network of interacting computational and physical devices"* [10]. Developing such complex systems are extremely challenging to complexity and strict regulated standards and guidelines. One example could be complexity comes when everything getting smart, e.g., phones, houses, cars, aircrafts, factories, cities etc. On the other, hand complexity in our systems are not visible to user and package as making life easier feature. In case of a smart car, the complexities such senor inter-communications, with operating system and internet connection is hidden from a driver.

CPS constitute of the physical part (hardware and its integration) and software with capability exposure as services *(services on-device, i.e. running within the CPS, and in network, i.e. other services running outside the CPS such as the cloud or even in other CPS* [10]. It is evident that binding force of today's CPS components is software. It is the core for development and maintaining smart environment where systems are interconnect intelligently. Agile approaches are widely used in classical software and systems development due to various benefits such as reduce time to market, better meet customer requirements, improve software quality, deliver value added feature to customer and many more [11, 12]. However, directly adopting Agile methods to CPS development is not explored and Lee [13] propose a research direction to "*Rethink the hardware and software split*" and its development. This leads to a question, what are the techniques are there to develop and maintain software for CPS. There are attempts to use Agile methods in the development of various CPS. Cawley et al. [14] investigated lean and agile methods in regulated, safety-critical systems; Huang et al. [15] reported positive experiences of Agile use in systems engineering for satellite development. Despite these attempts on Agile use in CPS, existing scientific literature seems to disperse and fragmented. This become the motivation for study to investigate the status of Agile methods in cyber physical systems development in existing literature. The goal of this systematic literature review is to identify the methods and practices used in the development of cyber physical systems, and faced challenges while using Agile in CPS as well as lessons learned along with future research direction.

The article is organized as follows: Sect. 2, Introduces the systematic review protocol in detail; Sect. 3, Reports the results of the review; Sect. 4, Discuss the validity threat; Sect. 5 conclude the paper with future research direction.

2 Research Method

In this section, we discuss our literature review protocol based on Kitchenham guidelines [6]. The review protocol has the following key elements: motivation and identification of research questions, Search Strategy, inclusion and exclusion criteria, quality assessment and data extraction.

2.1 Motivation and Research Question

The motivation and formulation comes under the umbrella of planning literature review phase. According to Kitchenham et al. (2011), "*classification and thematic analysis of literature on a software engineering topic*". Our motivation for this study is to provide a state-of-the-art of Agile methods use in the context of cyber physical systems between 2010 and 2018. The main objective is to identify and categorize Agile methods use in CPS development, distil the reported challenges of Agile methods in CPS, lessons learned and future research direction in the topic area. These broad objectives can be achieve with the following research questions.

- RQ 1. What is the status of applying agile methods use in context of CPS development?
- RQ 2. What are the challenges involved in Agile use for CPS development?
- RQ 3. What are the lessons learned from existing literature for the industry and the research community?

2.2 Search Strategy and Selection

The search string for this study was based Kitchenham et al. [6] guidelines-define 'population' and 'intervention'. *Population* refers to the application area, which is software, and *intervention* is Agile. Software is the expected search that will include all documents with the word "software" in title, abstract or keyword [3]. The search string was "(software AND (Agile OR Scrum) AND "cyber physical systems")". The rationale for using the term "software" is that, this study will cover studies that discuss software, software development, or software intensive products, services, and systems including cyber physical systems.

We chose four major databases in the field to search for relevant literature: Scopus, IEEE Xplorer, Web of Science, ACM digital Library. The retrieved papers from these databases are listed in the following Table 1.

The mentioned databases resulted into 101 papers. In step 1, was to sort the paper in one excel sheet and remove the duplicates. In total 32 duplicate papers were removed. The second step was to examine papers based on titles. Paper is exclude, if it clearly showing that they were outside the focus of this study. However, excluding decisions were not in all cases based only on titles, as some of the titles did not reveal the contents of the papers clearly enough. In such cases the papers were included for review in the next step.

In third step, each paper abstract, keywords are examine against the inclusion and exclusion criteria; which resulting into 38 papers exclusion. The remaining 31 papers were left for full reading. This fourth step yielded into 15 primary studies.

2.3 Inclusion and Exclusion Criteria

Studies were eligible for inclusion in this review if they are: written in English, published between 2010–2018, reported on applying the agile method or practices to CPS software development; argue for or against using any agile method or practices when developing CPS. The paper were excluded if their focus was not specifically Agile

Table 1. Selected databases and retrieved papers.

Databases	Full Query with filter	No. of retrieved papers
Scopus	TITLE-ABS-KEY (software AND (agile) AND “cyber physical systems”) AND (EXCLUDE (PUBYEAR, 2019)) AND (LIMIT-TO (LANGUAGE, “English”)) AND (EXCLUDE (DOCTYPE, “ch”)) AND (EXCLUDE (SRCTYPE, “k”))	35
Web of Science	(software AND (Agile) AND “cyber physical systems”) Refined by: LANGUAGES: (ENGLISH) AND [excluding] PUBLICATION YEARS: (2019) Timespan: All years. Indexes: SCI-EXPANDED, SSCI, A&HCI, CPCI-S, CPCI-SSH, ESCI.	20
IEEE Xplorer	software AND (Agile) AND “cyber physical systems” Filters Applied: Conferences, Journals	26
ACM Digital Library	Searched for software AND (Agile OR Scrum) AND “cyber physical systems” Published since: 2010 Published before: 2018	20
Total		101

methods in the context of CPS, short articles, duplicate papers, simulation studies, not peer reviewed papers.

3 Results

We identified 14 primary studies on Agile methods use in cyber physical systems domain (see Appendix A). From the year 2010 to 2013, no studies were identified which is dealing specifically with Agile in cyber physical systems development. Although the search process bring some papers published in that timeframe about Agility, however, they were related to systems technical agility and responses or algorithmic agility. The following Table 2 highlighted the increase in number of papers on Agile use in CPS development.

Table 2. Primary papers published annually

Years	2010–13	2014	2015	2016	2017	2018
Papers	–	1	3	3	7	1
Percentage	–	6.66%	20%	20%	46.66%	6.66%

Almost half of primary studies (46.66%) were published in year 2017, which can be considered as indicator of growing interest in using Agile method in cyber physical systems development and reporting about it. Furthermore, the primary studies mostly published in conferences, most of which are experienced based and qualitative studies (See Table 3).

Table 3. Paper distribution according to publication channel and occurrence

Source	Type	No. of papers
International Workshop on Rapid Continuous Software Engineering	Conference	2
International Conference on Computer Science and Software Engineering	Conference	1
International Conference on Parallel and Distributed Systems	Conference	1
Proceedings of the IEEE	Conference	1
Annual Conference of the IEEE Industrial Electronics Society	Conference	1
International Conference on Software Engineering: Software Engineering Education and Training Track	Conference	1
International Conference on Cyber-Physical Systems, Networks, and Applications	Conference	1
Cluster Computing	Journal	1
Americas Conference on Information Systems	Conference	1
CEUR Workshop Proceedings	Conference	1
Proceedings of NordDesign	Conference	1
International Conference on ICT Management for Global Competitiveness and Economic Growth in Emerging Economies	Conference	1
Ambient Intelligence - Lecture Notes in Computer Science	Conference	1
Annual Conference of the IEEE Industrial Electronics Society	Conference	1
Total		15

The primary studies didn't explicitly shows which Agile methods are dominantly used for CPS development. However, our analysis exhibits that Scrum practices dominantly reported. The primary studies are further categories based on themes as shown in the following Table 4. At higher level, two categories were identified. The categories are based on the number of primary studies discuss a specific theme.

Table 4. Primary studies categorization

#	Agile use theme wise in CPS	Primary studies
1	Development	P1, P2, P5, P8, P9, P11, P14, P12
	Requirements and Testing	P4, P6
	Continuous Deployment	P3
2	Teaching Agile in CPS engineering curriculum	P7, P10, P13, P15

Agile methods are well known and widely used in the software industry; while companies developing software cyber-physical systems hesitant to adopt such processes [P1, P2]. Wagner [P1] highlighted that in software development the final design is product, however, in hardware and mechanics it is the starting point of production. There are several proposed ways to implement Agile in CPS development. For example Wagner [P1] propose Scrum extension Scrum CPS. The extension helps to handle concurrent hardware and software development (See Fig. 1, Adopted from [P1]).

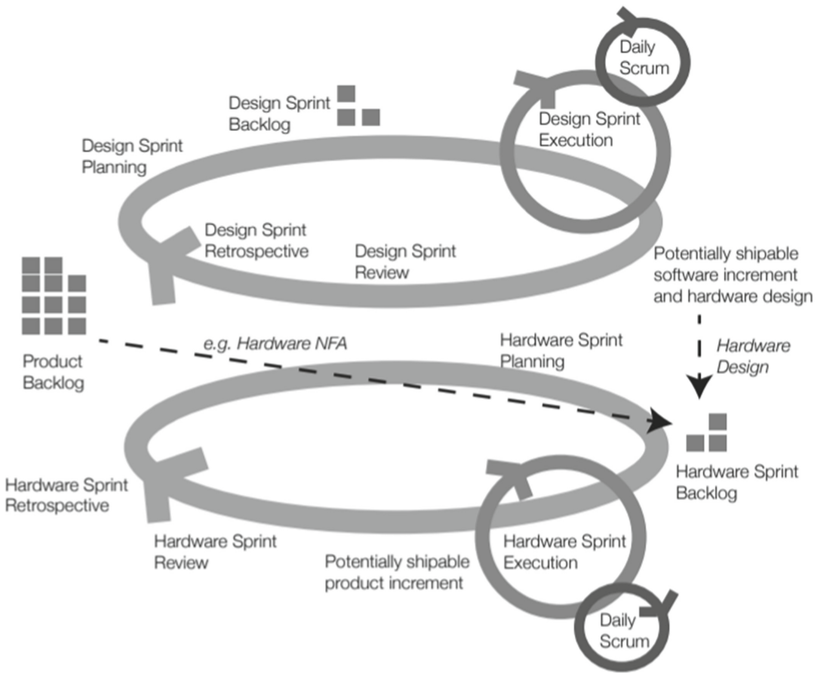

Fig. 1. Wagner propose Scrum CPS process [P1]

The central idea is to bring harmony between software and hardware development. In this regard, starting sprint could be the architecture of CPS and build operational software. The hardware design is a central part of software development backlog. The developed software going to be tested together with hardware simulations from design models, which build in parallel [P1]. One proposal is to adapt the idea of Agile Release Train from Leffingwell model [4], to ensure sprints synchronization and produce coherent CPS. It is important to bring closer and empower the development and hardware teams to collaborate with customer openly.

A single contact from the CPS team to customer can bring hurdle to process of development that is why to split product owner responsibilities to a number of persons [P2, P8]. For iterative development, Buchmann and Karagiannis [P9] experiment Agile modelling. Further, Pfeiffer et al. [P5] propose to create ad hoc personas to obtained global view of users. These personas should be iterative in nature in order to refine it based on data gathered from user research and make it more real. The proposed persona approach was adopted from Pruitt and Adlin [5].

Testing CPSs makes are extremely prone to human error due to complex modelling [P4]. Spichkova et al. [P4] propose a human-centered agile modelling and testing approach (HCATD) for CPSs. HCATD helps to increase efficiency of testing of CPSs and reduce the testing cost and time. The basic behind HCATD is combinatorial test design-test-planning technique. Further, functional requirements for reconfigurable cyber-physical system is challenging task. Boschi et al. [P6] described a methodology based on the requirement engineering concepts. It helps to in identification and deployment of general business and strategic requirements needed to implement new plug-and-produce paradigms into traditional production systems.

Continuous deployment in mission-critical CPS should be reliable and failure tolerant, where failure recovery is timely and predictable, and minimizes the negative impact on quality of service [P3]. A number of continuous deployment techniques exists for the CPSs, such as Locality-Enhanced Deployment and Configuration Engine (LE-DAnCE), Lightweight CORBA Component Model, a Methodology for Architecture and Deployment of Cloud Application Topologies based on iterative refinement (MADCAT), Dynamic IoT Application Deployment (DIANE) [P3]. To handle the CPS complexity, synchronization, failure detection, and state preservation, a solution could be microservices architecture [P3]. Microservices architecture helps to independently deployment of service on its own resources, scalability that are manage at runtime.

3.1 Teaching Agile in CPS Engineering Curriculum

Preparing student for the fast growing ICT industry is important. The engineers requires multi-disciplinary technical and soft skills. It is essential for software engineering and other ICT students to know about Agile development, model-driven engineering, and cyber physical systems [P7, P10]. Various universities included the Agile software development in their engineering education programmes. However, the primary studies highlighted that directly using Agile methods in other engineering domains are challenging and it need to be adapted [P7, P10, P13]. Marusik et al. [P10] proposed a task-centric holistic agile teaching approach. This approach utilized the problem/project based teaching and encourage students to do research in order to understand and adapt specific techniques in given situation. Furthermore, Mäkiö et al. [P15] stress more to include social and other software skills development in the education, which will result in well-prepared engineers for future jobs. This approach is similar to the software factory projects and courses in the field of software engineering [7, 8].

3.2 Challenges in CPS Development

The following challenges are reported in the primary studies while using Agile methods and practices in CPS development:

- CPS are complex systems, and to handle the complexities of highly networked and real-time components is difficult [P1, P11].

- The CPS is a combination of software and hardware development and synchronization of both is bigger challenge [P1, P11, P14].
- Insufficient visibility and control to efficiently manage and track software releases and project progress [P2].
- The software and hardware developers use different vocabulary, which makes it difficult to understand the cross-functional design issues [P11].
- CPS dependability is a major concern. It is measures in term of availability, reliability, durability, safety and security of a system. For example, automotive standard ISO26262, impose a rigorous process on development team, which bring difficult to implement Agile methods [P11].
- Critical system has a long lifespan and their maintenance are in form of patches, updates and repairs will form a major expenditure and requires more well advanced planning [P2]. Additionally, non-portability and lack of robustness will become easily suicidal for the system [P2].
- Need for complete deployment notations to allow stakeholders specifying and visualizing large-scale deployments from different perspectives and levels of abstraction [P3].
- A need for notation and tool support for linking design and runtime deployment concept [P3].
- Tool support for the evolution of deployment specifications and configuration management at runtime [P3].
- During the designing of CPS, developers do not take in consideration time to market [P11].
- Mismatches in the abstraction levels of the different components [P2, P11].

3.3 Lessons Learned

During the implementation of Agile methods and practices for CPS development, the following lessons learned are reported in the primary studies:

- Adapt agile software development principles and practices into real, every-day used, well and thoroughly understood and accepted ways of working, is hard work [P2, P14].
- Establish a dual-track requirement handling process to handle false assumptions about a feature [P2].
- Split the product owner responsibilities to a number of persons in team, which will help to have effective, communicate with both the customer and the teams [P2].
- The teams should have direct access to a real customer (or at least a legitimate customer proxy). Customer should proactively help the team to pick value added items for backlog [P2].
- Organization should aims to enable the mechanisms for the continuous improvement [P2].
- CPS emphasize the connectedness of many quite different components. To some degree, the availability of models for simulations and in-the-loop tests helps to test these aspects. In addition, the Agile Release Train ensures compatible interfaces between components connected via networks [P1].

- Agile modelling satisfy requirements from the client IoT application [P9].
- Show working system to customer at regular intervals - early system-level evaluation; which will help to avoid wrong design and development assumptions [P11, P14].
- Optimize the processes to enable shorter time-to-market, for example performing independent tasks concurrently [P11, P14].

4 Threats to Validity and Reliability

In this study, Wohlin et al. [16] guidelines were followed, to evaluate validity threats (i.e. construct validity, external validity, external validity, and conclusion validity). Firstly, construct validity meaning obtaining the right measures for the concept being studied. This threat was reduced with the help of data collection process presented by Pilar et al. [3] and apply clearly sorted out the inclusion and exclusion criteria's. Secondly, external validity which is related to generalizability of study results. To enhance the generalizability to this study results, data collection procedures were carefully executed and context of each primary study was examined. Thirdly, Internal validity examine the causal relationships of investigated factors. This literature review doesn't aim to examine or establish statistical causal relationship of the topic; hence this threat is not considered in this study. Fourthly, conclusion validity, meaning researcher biasness during interpretation of data. This is tricky and cannot be fully eliminated. However, measure were taken to reduce the researcher biasness by keeping full trail of data collection till obtaining 15 primary studies. Furthermore, the publications were obtained only from four databases Scopus, IEEE Xplorer, Web of Science, ACM digital Library. It is claimed that these databases has most of the relevant scientific article and used in similar literature review studies, such as Pilar et al. [3].

5 Conclusion

This literature review was aiming to analyze the current usage of Agile methods in cyber physical systems development, along with its challenges and lesson learned. The review exhibit that currently there is a lack of rigor in scientific literature on Agile methods use for cyber physical systems development. However, this study analysis provide valuable information for further investigations.

The results show that agile methods and practices are used at abstract level for CPSs development. Not a single industrial study clearly demonstrated that how Agile methods can be implemented. Only the Wagner [P1] proposed Scrum CPS process provide a bigger picture, but with limited implementation. However, the publication trend of primary studies shows that there is a growing interest in the use of Agile methods for CPSs development. The main reason is the iterative development and active customer role to reach the market on time. It also helps to build crossover team to design and build the systems collaboratively.

A number of challenges were highlighted in primary studies, such as; combining the software and hardware developers is difficult to the use of different vocabulary. The highly regulated CPSs requires following a variety of international standards that slow down the use of Agile implementation. Organizations needs to understand that being Agile is a continuous journey, which requires commitment and active role of all stakeholders. There is no silver bullet and CPS development teams requires optimizing processes based on their needs and project requirement.

To prepare graduate for fasted growing ICT domain, the educational institutes need to offer multi-disciplinary courses. The current hardware and software developers find it challenge to communicate efficiently due lack of common vocabulary. Additionally, soft skills including teamwork and collaboration need to be integrated to such engineering teaching programs.

The current literature does not provide reliable and depth of Agile use for CPSs development, however, it shows a positive tendency toward using the Agile methods. The future research direction could be replicate the some of primary studies as well as this literature review adding forward and backward snowballing.

Appendix A – Primary Studies

P1. Wagner, S.: Scrum for cyber-physical systems: a process proposal. In Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering, pp. 51-56. ACM. (2014).
P2. Koski, A., Mikkonen, T.: Rolling out a mission critical system in an agilish way. reflections on building a large-scale dependable information system for public sector. In 2015 IEEE/ACM 2nd International Workshop on Rapid Continuous Software Engineering, pp. 41-44. IEEE. (2015).
P3. Jiménez, M., Villegas, N. M., Tamura, G., Müller, H. A.: Deployment Specification challenges in the context of large-scale systems. In Proceedings of the 27th Annual International Conference on Computer Science and Software Engineering, pp. 220-226. IBM Corp. (2017).
P4. Spichkova, M., Zamansky, A., Farchi, E.: Towards a human-centred approach in modelling and testing of cyber-physical systems. In 2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS), pp. 847-851. IEEE. (2015).
P5. Pfeiffer, T., Hellmers, J., Schön, E. M., Thomaschewski, J. Empowering user interfaces for Industrie 4.0. Proceedings of the IEEE 104(5), 986-996 (2016).
P6. Boschi, F., Zanetti, C., Tavola, G., Taisch, M.: Functional requirements for reconfigurable and flexible cyber-physical system. In IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, pp. 5717-5722. IEEE. (2016).
P7. Ringert, J. O., Rumpe, B., Schulze, C., Wortmann, A. Teaching agile model-driven engineering for cyber-physical systems. In Proceedings of the 39th International Conference on Software Engineering: Software Engineering and Education Track, pp. 127-136. IEEE Press. (2017).

P8. Scheuermann, C., Verclas, S., Bruegge, B.: Agile factory-an example of an industry 4.0 manufacturing process. In 2015 IEEE 3rd International Conference on Cyber-Physical Systems, Networks, and Applications, pp. 43-47. IEEE. (2015).
P9. Buchmann, R. A., Karagiannis, D.: Domain-specific diagrammatic modelling: a source of machine-readable semantics for the Internet of Things. Cluster Computing 20(1), 895-908 (2017).
P10. Mäkiö-Marusik, E., Mäkiö, J., Kowal, J.: Implementation of task-centric holistic agile approach on teaching cyber physical systems engineering. Twenty-third Americas Conference on Information Systems, Boston (2017).
P11. Denil, J., Salay, R., Paredis, C., Vangheluwe, H.: Towards agile model-based systems engineering. In CEUR workshop proceedings, pp. 424-429. (2017).
P12. Luedeke, T. F., Köhler, C., Conrad, J., Grashiller, M., Sailer, A., Vielhaber, M.: CPM/PDD in the context of design thinking and agile development of cyber-physical systems. DS 91: Proceedings of NordDesign 2018, Linköping, Sweden, 14th-17th August 2018. (2018).
P13. Mäkiö-Marusik, E., Mäkiö, J.: Implementation of task-centric holistic agile approach on teaching cyber physical systems engineering and distributed software development. ICTM 115. (2016).
P14. Wessling, F., Gries, S., Ollesch, J., Hesenius, M., Gruhn, V.: Engineering a cyber-physical intersection management–an experience report. In European Conference on Ambient Intelligence, pp. 17-32. Springer, Cham. (2017).
P15. Mäkiö, J., Mäkiö-Marusik, E., Yablochnikov, E., Arckhipov, V., Kipriianov, K.: Teaching cyber physical systems engineering. In IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society, pp. 3530-3535. IEEE. (2017).

References

1. Jasperneite, J.: Was hinter begriffen wie industrie 4.0 steckt. Comput. Autom. (2012). www.computer-automation.de/steuerungsebene/steuern-regeln/artikel/93559/0/
2. Hermann, M., Pentek, T., Otto, B.: Design principles for industrie 4.0 scenarios. In: 2016 49th Hawaii International Conference on System Sciences (HICSS), pp. 3928–3937. IEEE (2016)
3. Rodríguez, P., Haghighatkhah, A., Lwakatare, L.E., Teppola, S., Suomalainen, T., Eskeli, J., Oivo, M.: Continuous deployment of software intensive products and services: a systematic mapping study. J. Syst. Softw. **123**, 263–291 (2017)
4. Leffingwell, D.: Agile Software Requirements. Lean Requirements Practices for teams. Programs, and the Enterprise. Addison-Wessley, Boston (2011)
5. Pruitt, J., Adlin, T.: The persona lifecycle: keeping people in mind throughout product design. Elsevier (2010)
6. Kitchenham, B.: Guidelines for performing systematic literature reviews in software engineering, version 2.3, EBSE technical report EBSE-2007-01, Keele University and University of Durham (2007)
7. Ahmad, M.O., Liukkunen, K., Markkula, J.: Student perceptions and attitudes towards the software factory as a learning environment. In: 2014 IEEE Global Engineering Education Conference (EDUCON), pp. 422–428. IEEE (2014)

8. Taibi, D., et al.: “Free” innovation environments: lessons learned from the software factory initiatives. In: IARIA (2015)
9. Rajkumar, R., Lee, I., Sha, L., Stankovic, J.: Cyber-physical systems: the next computing revolution. In: Design Automation Conference, pp. 731–736. IEEE (2010)
10. Leitão, P., Colombo, A.W., Karnouskos, S.: Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges. Comput. Ind. **81**, 11–25 (2016)
11. Dybå, T., Dingsøyr, T.: Empirical studies of agile software development: a systematic review. Inf. Softw. Technol. **50**(9–10), 833–859 (2008)
12. Rodríguez, P., Markkula, J., Oivo, M., Turula, K.: Survey on agile and lean usage in Finnish software industry. In: Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, pp. 139–148. IEEE (2012)
13. Lee, E.A.: Cyber-physical systems-are computing foundations adequate. In: Position Paper for NSF Workshop on Cyber-Physical Systems: Research Motivation, Techniques and Roadmap, vol. 2, pp. 1–9. Citeseer (2006)
14. Cawley, O., Wang, X., Richardson, I.: Lean/agile software development methodologies in regulated environments – state of the art. In: Abrahamsson, P., Oza, N. (eds.) LESS 2010. LNBIP, vol. 65, pp. 31–36. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16416-3_4
15. Huang, P.M., Darrin, A.G., Knuth, A.A.: Agile hardware and software system engineering for innovation. In: 2012 IEEE Aerospace Conference, pp. 1–10. IEEE (2012)
16. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Experimentation in Software Engineering. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29044-2