Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 45 (2020) 313–318

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

10th Conference on Learning Factories, CLF2020

Bender – An Educational Game for Teaching Agile Hardware Development

Daniel Omidvarkarjan[a,*], Jonas Conrad[b], Constantin Herbst[b], Christoph Klahn[a], Mirko Meboldt[b]

[a]*Inspire AG, Technoparkstrasse 1, 8005 Zurich, Switzerland*
[b]*Product Development Group Zurich pd|z, ETH Zurich, Leonhardstrasse 21, 8092 Zurich, Switzerland*

Abstract

Within this paper, an educational game is presented that transfers Agile principles for the development of physical systems. The training leverages elements of Learning Factories (LF) to simulate an Agile hardware development project within two days. By doing so, the challenges of applying Agile within the hardware domain are realistically reflected. The training revolves around a physical wire bending machine, which a development team of four participants needs to modify within a realistic engineering and production setting. A trial with mechanical engineering students was conducted to validate the training design. The participants showed a positive attitude towards the active learning approach. Furthermore, the students expressed that they perceived the game to improve their learning regarding Agile hardware development.

© 2020 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of the scientific committee of the 10th Conference on Learning Factories 2020.

Keywords: Agile for hardware; Agile training; learning factory concepts; development simulator; Agile hardware development

1. Introduction

New Product Development (NPD) is confronted with a multitude of challenges, popularly summarized under the acronym VUCA. The term describes the rise of volatility, uncertainty, complexity and ambiguity [1]. In this context,

* Corresponding author. Tel.: +41 77 9232440.
E-mail address: omidvarkarjan@inspire.ethz.ch

2351-9789 © 2020 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of the scientific committee of the 10th Conference on Learning Factories 2020.
10.1016/j.promfg.2020.04.023

Agile has been identified as promising approach to overcome those challenges due to its responsiveness to change [2]. While Agile has already proven to be very effective in the software domain [3], hardware firms still struggle to fully exploit the approach. Prior studies [4] have identified that inadequate and dysfunctional trainings are one of the key issues hindering the adoption of Agile in industry. In this paper, we present a novel training module that transfers Agile principles by realistically simulating a hardware development project with elements of Learning Factories (LF).

2. Related work & research gap

In the field of Agile trainings, there has been a great effort to move from lecture-based modules to more involving learning formats. Examples are the LEGO based game by Paasivaara et. al. [5] and the SCRUMIA training by Wangenheim et. al. [6], which both have a strong focus on teaching concrete methods like Scrum with their specific roles and rules. To do so, the game environment is strongly simplified by using LEGO bricks, playing cards or paper. While this approach ensures that participants can focus on applying the method correctly, an absence of any technical challenge fails to transfer engineering or production related Agile practices. In addition, the abstracted development environment does not reflect real world industrial practice, as roles and tasks employed in the presented trainings correspond poorly to real-world organizations. To sum up, the presented trainings do not reflect engineering challenges adequately, or only to a limited extent. It is therefore unclear how well those trainings contribute to the application of Agile for hardware in real world applications.

The goal of this paper is therefore to present a teaching concept that transfers Agile principles while adequately reflecting the engineering challenges specific to the hardware domain. Furthermore, a first assessment with mechanical engineering students is presented. The evaluation considers the participants' reaction & perceived learning.

3. Learning goals

Table 1 shows the Agile principles that are covered by this training. In accordance to [7], principles are defined within this study as key characteristics that a process must reflect to be considered Agile. They have been derived from prior studies [7,8] and have carefully been adjusted throughout the iterative development of the training to fit the constraints of the game. In accordance to Bloom's taxonomy [9], the objective of this training is to cover the first three layers of cognitive learning levels, enabling the participants to remember, understand and apply Agile principles after completing the module.

Table 1. Learning goals of the game.

No.	Agile principle
1	Frequent interactions
2	Test-driven development
3	Self-organizing teams
4	Iterative progression
5	Continuous improvement
6	Customer involvement
7	Accommodating change
8	Simplicity

4. Game description

In order to realistically simulate the application of Agile in a hardware development project, the training is built around elements of Learning Factories (LF). The term refers to learning environments that mimic real production processes to develop competencies of present and future industry personnel [10]. The training's main structure is presented using the key characteristics of LF (see Figure 1): product, process, didactics and setting [11].

Dimension	Characteristic	Realization in training
Product	physical	wire bending machine
Process	authentic, technical & organisational	planning, design, fabrication & assembly, distributed to roles
Didactics	formal and informal learning, trainee actions	alternation of action & reflection, game mechanics
Setting	real & on site	production environment: workshop, laser cutter, CAD
Purpose	training & research	Agile training & research on knowledge transfer

Fig. 1. Overview of the training structure (adapted from [11]).

4.1. Product

The training revolves around a modular wire bending machine, that in its initial state can bend thin metal wire into two-dimensional forms (depicted in Figure 2a). The highly modular machine was specifically developed for the purpose of this teaching module. It is made from laser cut wood sheets, several machined parts and electronic components. During the training, the task is to modify the machine so that it can bend predefined three-dimensional structures. Therefore, the team must redesign the bending mechanism. The machine is designed in such a manner that only one reasonable solution is possible (depicted in Figure 2b), ensuring control over the game play for the moderators.

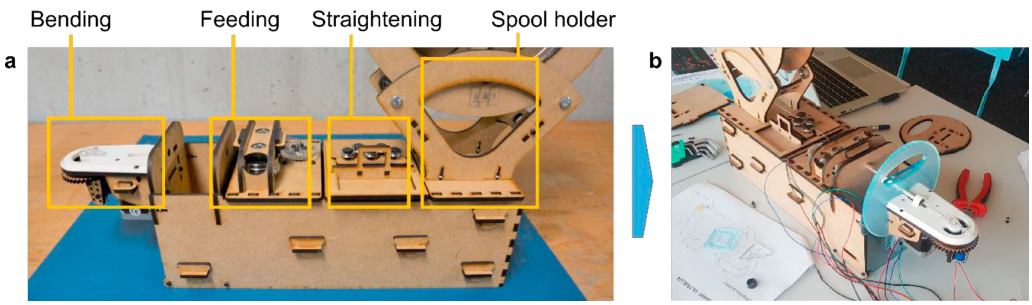

Fig. 2. (a) Initial wire bending machine; (b) Finished state of the machine (with modified parts in blue).

4.2. Process

The modification is conducted by planning, designing, fabricating and assembling new machine parts as well as creating code sequences to control the motors in order to produce the demanded wire structure. As illustrated in Figure 3, the participants are provided with two development sprints, lasting 100 minutes each, during which they can freely design, build and test their modifications. Both sprints are followed by delivery phases to produce the demanded wire structures and present them to a customer to collect feedback. The round is concluded with a retrospective session where the coaches provide methodical feedback based on observations that have been made during the development phase.

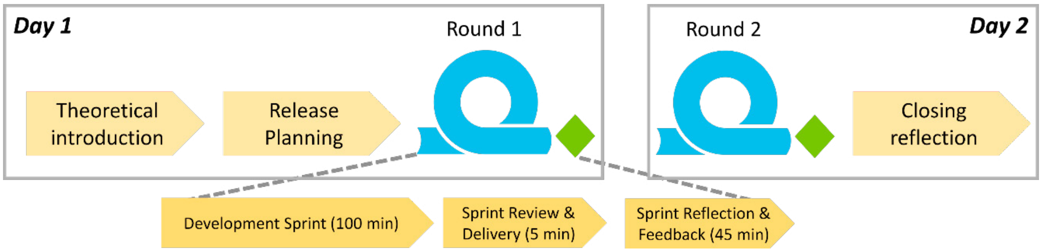

Fig. 3. Training schedule.

4.3. Setting

The training is conducted in a creative working space that houses provisions for the visualization of ideas, mechanical tool trolleys, a laser cutter and computer aided design workstations, mimicking an industrial development, production and assembly environment. Four different game roles are applied, representing the disciplines that one can typically find in engineering teams. Every role has its own set of responsibilities, tools and documentation (see Table 2). As the training aims to transfer Agile principles regardless of a specific method, no explicit Agile roles are used. Special focus was set on creating an environment resembling realistic conditions while simultaneously requiring minimal prior knowledge of the participants. As a result, the participants are provided with intuitive tools like a drag & drop CAD design system and a web-based programming interface for controlling the mechatronic components.

Table 2. Role description.

Role	Responsibilities	Tools
Mechanical design	Design of laser cut parts	Drag & drop CAD Tinkercad (www.tinkercad.com)
Manufacturing & assembly	Preparation and execution of laser cut jobs, assembly of mechanical parts	Tool trolley, laser cutter
Programming	Coding of jobs for bending machine	G-code equivalent, web based graphical user interface
Enabling & testing	Facilitation of team collaboration through methodical input	Methodological toolbox

4.4. Didactics

The training makes use of serious game mechanics, such as tutorials, role play, resource management and feedback. These can be linked to specific learning mechanics [12,13] to achieve experience or knowledge transfer during learning situations [10]. Two examples are elaborated to illustrate the link between game design & learning situations:

- Essential information about the machine and the task is spread between the different roles. The participants are therefore required to share role-specific information in order to solve the development challenge, promoting the learning goals *Frequent interactions* and *Self-organizing teams*.
- A limited number of production runs, in combination with a failure-prone design task, require prototyping to prevent errors and a considerate use of resources. This also reflects real-world industrial constraints, encouraging learning about *Test-driven development* and *Simplicity*.

5. Evaluation

To validate the training design, a trial was conducted with mechanical engineering students from both undergraduate and graduate level. The students had some prior knowledge on Agile, as they received an introduction to the topic within another university lecture. The goal of the exploratory evaluation was primarily to get an understanding how the participants react to the module and to test the robustness of the training in practice. In total, two training sessions with a total of 7 participants were performed. In accordance to the Kirkpatrick framework for assessing educational modules [14], the students' reaction and perceived learning were evaluated. After completing the training, the students were asked to submit written feedback by stating their thoughts towards the overall training design and their perceived learning regarding Agile principles. The following results are based on the qualitative analysis of their statements.

5.1. Reaction

Based on the given feedback, the overall impression of the training was generally positive. The students reported that they enjoyed the practical learning approach as it was challenging to them and created some sense of fulfilment after completing it. The development task was perceived as exciting and appropriate regarding level of difficulty. Some participants complained that the ideal solution was too obvious to them. They wished for more variety in possible solutions. The students liked the team-driven structure of the training. They appreciated the constant need for collaboration and communication between the different team members. The participants reported that they liked the setting of the training and perceived it as realistic. According to the students, the workshop space encouraged a test-driven approach and led to a good atmosphere. Some students wished for quick fixes for technical problems that are not primarily connected to Agile learnings (e.g. in the form of intervening instructions or hardware parts).

5.2. Learning

Apart from their reaction, the students were also asked to comment on their perceived learnings (depicted in Figure 4). The items were assigned to the game's intended learning goals (see Section 3) through clustering.

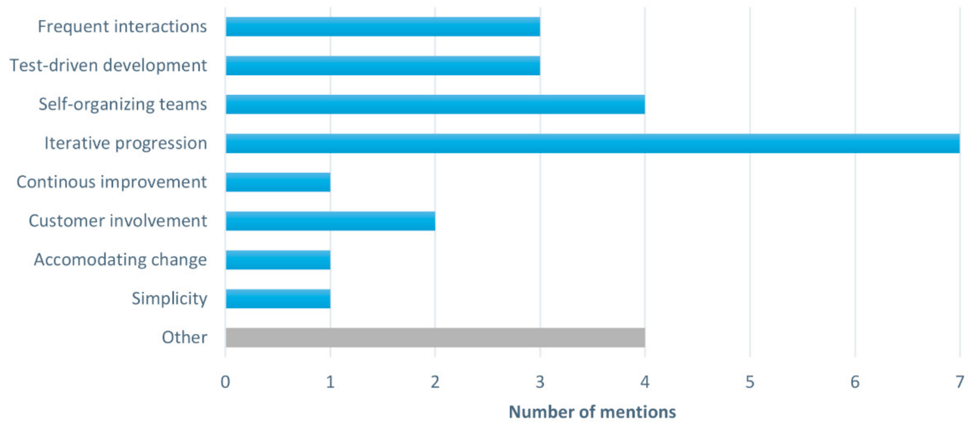

Fig. 4. Distribution of the reported learnings.

The learning goals have been reported by the participants with varying frequencies. The principle *Iterative progression* was mentioned the most by the students. Within this topic, major learning outcomes evolved around practices that enable the incrementalization of product features and the different level of iterations (whole product iteration versus sub sprints). Another principle that has been frequently reported by the students was *Test-driven development*. Here, the focus was mainly around how to abstract or simplify test cases to enable early and frequent validations under the constraints of physical systems. The teamwork related principles *Self-organizing teams* and *Frequent interactions* were also discussed in detail by the students. They reflected on their learnings regarding Agile leadership and the strength of cross-functional teams. In contrast to the aforementioned items, the following topics were mentioned less frequently: *Customer involvement, Accommodating change, Continuous improvement* and *Simplicity*. The low frequency of comments on these topics indicate that the students either did not learn about those items or did not perceive these aspects to be relevant to them. The participants also reported learnings that could not be assigned to any of the intended learning goals (*Other*). This included items like prioritization and time-boxing.

6. Discussion & Conclusion

Within this study, we presented a novel training module that transfers Agile principles by mimicking a hardware development project with elements of LF. The game-based training was tested with university students and evaluated regarding the participants' reaction and perceived learning. The learning outcomes incorporated both collaboration and engineering related Agile competences. Especially the transfer of engineering driven Agile principles such as *Test-driven development* is highly relevant, as the absence of these skills still act as the biggest hurdle hindering the widespread application of Agile in the hardware domain. In addition, existing trainings insufficiently cover these learning goals as they rely on abstracted training environments. The inclusion of LF elements was viable in this study as it enabled the participants to learn about Agile within the application context of hardware product development. The complexity of the mechanical system, together with the engineering process and setting, created situations mirroring real-life challenges in hardware development. The effect of proposed Agile countermeasures was experienced firsthand by the participants as seen in their reported learnings. This affirms the conceptual choices made in the game design and indicates that the LF elements improved the learning experience. To achieve this result, a finely tuned training design was required. The employed game task, rules, roles, infrastructure and documentation needed to be consistently aligned to another to achieve an appropriate learning environment that incorporates the complexity of hardware development without being exceedingly difficult to play.

In the future, we aim to investigate the use of quantitative evaluation tools to get a more profound understanding of the participants' reaction. Standardized questionnaires for the assessment of training modules could be employed (e.g. MEEGA+ [15]). The training could also be applied with other target groups such as industry participants. Furthermore, we are planning to conduct a follow-up evaluation to examine the long-term effectiveness of the training.

References

[1] N. Bennett, G.J. Lemoine, What a difference a word makes: Understanding threats to performance in a VUCA world, Bus. Horizons 57 (2014) 311–317.

[2] B.W. Boehm, R. Turner, Balancing agility and discipline: A guide for the perplexed, Addison-Wesley, Boston (2004).

[3] R.G. Cooper, A.F. Sommer, The Agile-Stage-Gate Hybrid Model: A Promising New Approach and a New Research Opportunity, J. Prod. Innov. Manag. 33 (2016) 513–526.

[4] T. Javdani Gandomani, H. Zulzalil, A.A. Abdul Ghani, A.B. Md. Sultan, R. Meimandi Parizi, The impact of inadequate and dysfunctional training on Agile transformation process: A Grounded Theory study, Inform. Soft. Technol. 57 (2015) 295–309.

[5] M. Paasivaara, V. Heikkilä, C. Lassenius, T. Toivola, Teaching students scrum using LEGO blocks, Proceedings of the 36th International Conference on Software Engineering ICSE 2014, May 31-June 07, New York, NY, 382–391.

[6] C.G. von Wangenheim, R. Savi, A.F. Borgatto, SCRUMIA—An educational game for teaching SCRUM in computing courses, J. Syst. Softw. 86 (2013) 2675–2687.

[7] A. Sidky, J. Arthur, S. Bohner, A disciplined approach to adopting agile practices: the agile adoption framework, Innovations. Syst. Softw. Eng. 3 (2007) 203–216.

[8] S. Soundararajan, O. Balci, J.D. Arthur, Assessing an organization's capability to effectively implement its selected agile method(s): An objectives, principles, strategies approach, 2013 IEEE Agile Conference, August 05-09, Nashville, TN, 22–31.

[9] B.S. Bloom, Taxonomy of Educational Objectives: The Classification of Educational Goals, Longman, New York (1956).

[10] M. Tisch, C. Hertle, E. Abele, J. Metternich, R. Tenberg, Learning factory design: a competency-oriented approach integrating three design levels, Int. J. Comp. Integr. Manuf. 29 (2016) 1355–1375.

[11] E. Abele, J. Metternich, M. Tisch, G. Chryssolouris, W. Sihn, H. ElMaraghy, V. Hummel, F. Ranz, Learning Factories for Research, Education, and Training, Procedia CIRP, 32 (2015) 1–6.

[12] S. Arnab, T. Lim, M.B. Carvalho, F. Bellotti, S. de Freitas, S. Louchart, N. Suttie, R. Berta, A. de Gloria, Mapping learning and game mechanics for serious games analysis, Br. J. Educ. Technol. 46 (2015) 391–411.

[13] P. Lameras, S. Arnab, I. Dunwell, C. Stewart, S. Clarke, P. Petridis, Essential features of serious games design in higher education: Linking learning attributes to game mechanics, Br. J. Educ. Technol. 48 (2017) 972–994.

[14] D.L. Kirkpatrick, J.D. Kirkpatrick, Evaluating training programs: The four levels, 3rd ed. Berrett-Koehler Publishers, San Francisco (2010).

[15] G. Petri, C. Gresse von Wangenheim, MEEGA+: A method for the evaluation of the quality of games for computing education, Proceedings of SBGames 2019, October 28-31, Rio de Janeiro, RJ, 1384–1387.