

COMBINING SCRUM AND DESIGN THINKING FOR A HIGHLY ITERATIVE AND USER-CENTRIC HARDWARE DEVELOPMENT PROJECT

Augustin, Laura; Schabacker, Michael

Otto-von-Guericke University Magdeburg

ABSTRACT

User-centric development is essential to any product development project, especially in order to keep up with today's ever-changing product cycle.

This paper explores the potential of combining specific aspects of Scrum and Design Thinking to maximise user integration as well as implementing short iterations in hardware development projects, in this use case a three-month development project at a German high-end homeware manufacturer. In addition to observations regarding the application of those approaches this paper will then offer a range of newly developed user-centric methods to efficiently integrate the user's perspective in future development projects, as well as feedback from the product developers at the company and comparisons to current methods.

This use case was furthermore able to illustrate how the employment of such methods made short and sprint-like development cycles within hardware development attainable.

These user-centric methods developed within the use case can be applied to future application- and user-oriented projects in order to speed up the product development process and ensure that the product or service matches the users' needs and desires.

Keywords: User centred design, Integrated product development, Design methods, Scrum, Design Thinking

Contact:

Augustin, Laura
Otto-von-Guericke University Magdeburg
LMI
Germany
laura.augustin@ovgu.de

Cite this article: Augustin, L., Schabacker, M. (2019) 'Combining Scrum and Design Thinking for a Highly Iterative and User-Centric Hardware Development Project', in *Proceedings of the 22nd International Conference on Engineering Design (ICED19)*, Delft, The Netherlands, 5-8 August 2019. DOI:10.1017/dsi.2019.225

1 INTRODUCTION

User integration, while having been a widely researched topic in science since the 1970s and 1980s (von Hippel 1978, von Hippel 1988, Sandmeier 2008), still has not fully arrived at some of the most traditional industries. Market research often is the only way some companies try to integrate the user, as seen in the use case discussed in this paper. Scrum, though iterative in nature and therefore perfectly suitable for frequent user feedback, has been criticised for inadequate user integration (Singh, 2008).

The objectives within the use case were to implement Scrum's iterative approach in combination with user-centric aspects of Design Thinking within a hardware development project and to evaluate results as well as observations made throughout the process. In addition to observations during implementation, this paper will then offer a range of newly developed user-centric methods to efficiently integrate the user's perspective in future development projects as well as highlighting feedback from the participants involved.

2 MODEL SELECTION

Involving the user in the development process has shown to not only increase a product's success but also decrease development costs by focusing the development on the most important issues early on in the process (Seely, Brown and Hagel 2005). A multitude of different approaches have been developed to gain a greater insight into the users' wants and needs while integrating them in the process (von Hippel 1978, von Hippel 1988, IDEO 2015). Since Design Thinking was already in its first phase of implementation at the company in question, this approach was included in the use case.

Lean Startup is built on frequent user feedback within each development loop and creating a minimal viable product which is continuously improved through feedback loops (Ries, 2011). This approach is mostly applied to software development. With Scrum being an agile and flexible development model whose iterative approach favours frequent user integration, it was chosen for the use case at hand, especially since its rulebook clearly states, that it is applicable to software *and* hardware development (Schwaber and Sutherland, 2017).

2.1 Accelerated iterations with Scrum

Scrum's highly iterative approach ensures flexibility in the development process while constantly integrating user feedback (Schwaber and Sutherland, 2017). These iterations and the inclusion of user stories places users and their needs at the centre of the development process and ensures the product or service is always tailored to its target audience (Schwaber, 1995).

2.1.1 Early testing with prototypes

At the end of each sprint within the development project, a usable product is created. This so-called minimum viable product (Kusay-Merkle, 2018) is then tested by users and therefore produces feedback, which can be integrated into the next sprint. During a four-sprint development cycle for example, four user feedback loops are possible, instead of the traditional market research approach at the beginning and the end of a development cycle. There are, however, a few constraints in hardware development, that need to be considered.

2.1.2 Constraints in hardware development

With Scrum originating in software development, some adjustments need to be made in order to fully use its potential in hardware development. The most obvious but also most detrimental difference between the two is the need for physical prototypes in hardware development. Consequently, Scrum's time constraints might not work in every sprint, since waiting times for physical prototypes will be longer, simply due to the fact that not all prototypes may be built by the Scrum team itself.

Within the development project at hand these waiting times became most apparent during later development phases, where more detailed and polished prototypes had to be made. As soon as prototype production has to be moved outside the development team, it loses its independence, which is one of the most critical facets of Scrum. Without independence, strict time constraints cannot be kept.

Development sprints therefore need to be flexible in order to allow for delays in model making. Prototypes in the earliest sprints have to be simplified, i.e. made from cardboard, paper and office

supplies at hand. This will allow for early testing with rough models instead of creating long waiting times early on.

2.1.3 Transferring Scrum's highly iterative approach to hardware development

One key aspect of Scrum is its iterative approach. Applying it to hardware development will allow for early and steady feedback within the development process.

Collecting early user feedback in hardware development can be difficult. Especially within the first few development stages prototypes tend to be rough models, that focus on one aspect of the design. It will either be a functional model or a design study and rarely a combination of the two, at least not until later in the project. In order to understand a concept with only half of the information a certain level of imagination is necessary. This can be difficult within a development team, let alone a group of unbiased (but also uninformed) users. Often their feedback is concerned with unfinished aspects of the prototype ("Why is it made from this material?"), rather than just the aspect in question (i.e. functionality).

Therefore, the work was focused on alternative methods of gaining valuable insight into the user's wants and needs, especially during the earliest development phases. Rather than letting the user give feedback, the goal was to observe the user in a practical user scenario context and avoid direct questioning.

2.2 Empathic product development with Design Thinking

Empathy is at the core of the Design Thinking philosophy and always the catalyst for a development project (Doorley, Holcomb 2018), as shown in figure 1. Empathy also frames all following development phases in order to keep the user in mind every step of the way.

For example, instead of looking to define a list of requirements, the first order of action would be to analyse the user experience and feelings attached to the product. The starting point is to improve the experience rather than the product.

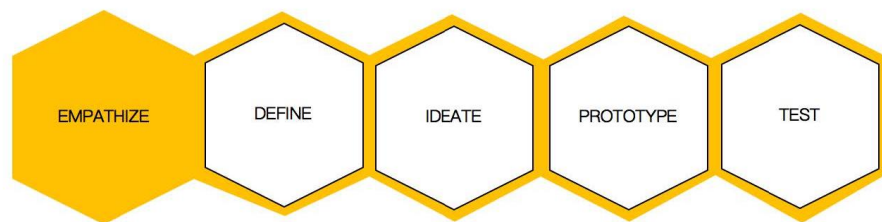


Figure 1. Empathy as a starting point, while being present in all development phases, based on d.school (Doorley, Holcomb 2018, p. 2)

Based on the work of Brenner and Uebornickel (2016), as well as Plattner and Meinel (2015), Design Thinking is looking to involve the user in as many aspects of the development process as possible. This means not only asking for feedback at the end of every development cycle, but also including the user in the sketching, ideating and model building process.

2.2.1 Utilising the user as part of the development team

In the Design Thinking booklet "The Field Guide to Human-Centered Design" (IDEO, 2015) methods are categorised into inspiration, ideation and implementation and are designed to help including the user's perspective as directly as possible.

The method "peers observing peers" (IDEO, 2015, p. 60) involves the users by letting them document their own routine, thus not only recording their actions but also giving them the opportunity to document it in their personal fashion. Therefore, two aspects from the user's perspective are documented, the 'what' and the 'how' (what they choose to document and in what manner it is done).

The "co-creation-session" (IDEO, 2015, p. 109) incorporates the target group in a workshop like environment in order to create models and rough prototypes together.

Both methods involve the user directly, rather than just conducting an interview. The user is part of the development team and can personally impact the process as well as the result.

This approach is especially useful, when the development project is fairly small and the user involved is directly affected by the resulting concepts. Product acceptance is increased naturally by giving the

user a valuable role in the process. This of course requires a great amount of time, which especially in short development projects is hard to come by.

2.2.2 Results and challenges of user integration in all development stages

To involve outside perspectives is above all a rather time-consuming endeavour including planning, execution and post-processing. Method catalogues like the Field Guide help to save time and resources by giving an overview of possible approaches, but the methods themselves are still relatively time consuming (several hours per method) (IDEO, 2015). Workshops take up a day or more, while also taking time in preparation and post-processing. There is no guarantee for results or useful output either. Hence weighing up expense and advantage is a key component.

To involve the user in early development stages can also be tricky, since early models will not look polished and therefore limit the user's imagination. It can be difficult to look at a cardboard model and envision the final product, as discussed in section 2.1.3. Furthermore, negative feedback is always easier to give from a user's perspective, since flaws of a prototype are easy to spot. It is important to ask the right questions and know how to interpret the answers. For example, it may be more useful to ask what needs to be improved rather than what is already working perfectly.

2.3 Synergies of Scrum and Design Thinking

Scrum and Design Thinking are both iterative, interdisciplinary and user-centric methods (Schwaber, 1995, Brenner and Uebernickel 2016), that focus on different approaches to user integration. Whereas Scrum ensures the development is centred around user feedback and its steady incorporation in each sprint (development for the user), Design Thinking focuses on involving the user in each development phase personally (development with the user). This is why combining the two bears such potential in a development context. The iterative structure of Scrum paired with the direct involvement of the user in the development process with Design Thinking enables the use case's development team to both ensure the product is made with the users in mind but also including them in the process.

The development project was designed to be a testing field for a wide range of methods and approaches on how to integrate the user directly into the development project while implementing Scrum's short development sprints.

3 DESCRIPTION OF THE USE CASE

The use case at hand was a product development project with a prestigious homeware manufacturer in Germany. The objective was to reinvent the process of loading and unloading the dishwasher with new concepts. The following sections are divided into the use case's given conditions, its structure, execution and evaluation.

3.1 Given conditions

The use case took place within the company's innovation & development department, whose given conditions need to be discussed in order to correctly assess the use case's course and its results.

3.1.1 Product development process

Product innovation takes place within the innovation & development department and is moved along to CAD and manufacturing specialists once a concept is approved and needs refining for production. Figure 2 is illustrating a simplified version of the development process within the innovation department. Usually one iteration from research to handover takes at least one year. The trigger for a new development project can be either input from market research or internal factors, for example continuous product improvement for certain products.

The grey arrows signify external factors to the development process, which product innovation cannot greatly influence. The dependency to market research is clearly visible and had to be avoided for the use case, since its timeframe was a quarter of its usual duration.

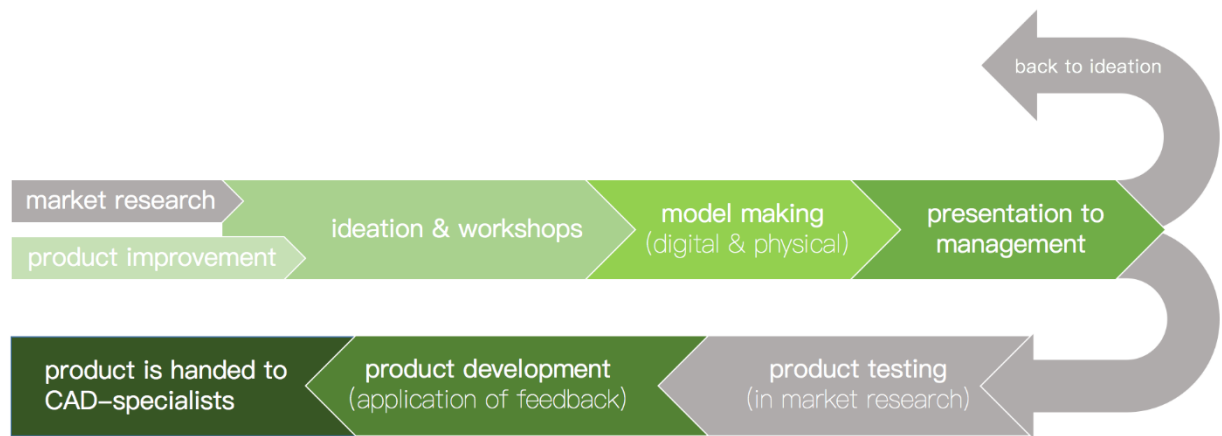


Figure 2. Development process at the innovation department

3.1.2 Previous methods of user integration

Within the product development and innovation department many approaches to user centric design were known, but far from being implemented in their daily routines. When a new development project is started, the main approach to integrate the user is to involve market research. Since market research is a completely different department, waiting times of up to three or four months are to be expected. Furthermore, the most commonly used tool is a questionnaire, which in itself is not the most ideal form of user integration, since it is very one sided and does not allow for follow up questions or spontaneity.

Other tools used were user panels to test final prototypes, or in-home visits to a group of chosen users. Both approaches were very time consuming and always realised by hired companies, which was also very cost-intensive. The development team itself had no means of internal user integration within the department. Ideation workshops only included colleagues, admittedly from different departments, but never users or customers.

3.1.3 Methods for iterative user integration

Methods employed by the development department are mostly ideation techniques, whereas all user research via questionnaires and user panels is fully conducted by the market research department. Since the project at hand had the very limited time frame of three months, involving market research did not make much sense, since its findings would come at the end of the project. Nevertheless, the department was asked to participate and as predicted, delivered results once the project came to a close. This meant that user integration had to be implemented in the development process in order to keep the schedule.

In this context Scrum was implemented to make a three-month development project possible as well as realising prototypes by the end of every sprint. The objective was to examine whether early model making within the first three weeks was possible and feasible in hardware development.

The Design Thinking approach was chosen to increase user integration within the product development process and to trial different methods from “The Field Guide to Human-Centered Design” (IDEO 2015).

3.2 Structure and execution of the use case

The use case’s timeline is illustrated in figure 3, including planned and true sprints. This was done in order to sufficiently analyse the implementation of the approach.

Sprints were planned separately during each sprint planning phase. Activities were defined before each sprint and were adapted when necessary. Whenever two activities were intertwined, they are striped in the corresponding colours of each activity within the sprint plan, as shown in figure 4.

Due to company-specific time management the first sprint’s presentation was moved up the schedule by three days. This meant that the originally planned model making did not fit within the sprint.

Consequently, the first sprint did not end with a minimum viable product. This could also be due to the fact that the product was too complicated to build so quickly and there was no way to simplify it enough so that early model making would make sense. Instead of models, detailed sketches were made in order to visualize the concepts.

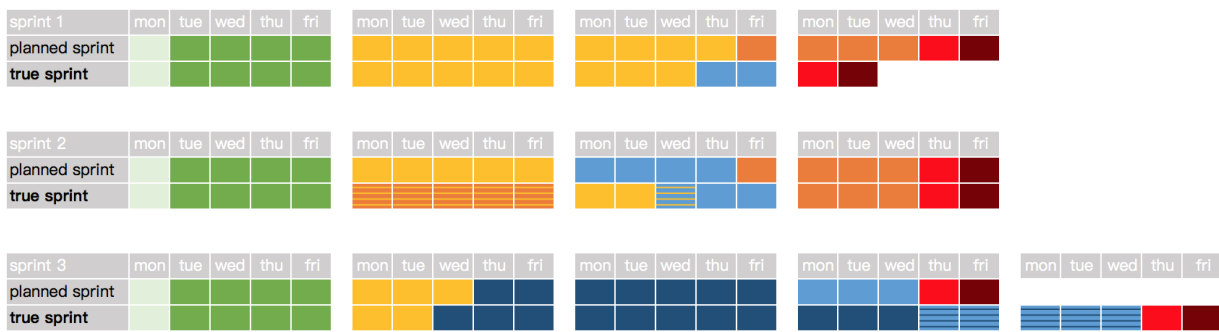


Figure 3. Sprint overview of planned and true sprints



Figure 4. Colour code of activities

The second sprint was mostly true to its plan, but its planned activities needed to be adapted. This is also the first sprint with mixed activities, ideation and model making were combined in order to help the ideation process move along. As a result, final model making at the end of the sprint could be cut down by a day.

As predicted, delays due to manufacturing periods did in fact prolong sprint duration. Sprint 3 deviated the most from its original plan and took five instead of the planned four weeks, since final prototyping was the focus of this sprint. Prototyping took over three weeks and therefore had to take place parallel to product visualization within the last week of the sprint.

Market research was asked to participate, but since planning their integration already took longer than the execution of the first sprint, meaning user feedback had to be implemented in the development process in order to keep the schedule. Market research delivered results after the third sprint was completed and therefore could not be used in this development project. This was mostly owed to the fact that within the usual development projects, market research generally had up to two to three months for planning and completion of their work. Of course this does not necessarily fit within Scrum's timeframe.

3.3 Observations regarding the implementation of Scrum's timeframe

The development project's structure was greatly impacted by Scrum's short timeframes for each sprint. Research, ideation, construction and prototyping all had to fit into a four-week sprint, which means that each activity could take about a week. Models that were built within the first sprint were very "rough" and as discussed earlier, this poses a challenge to integrate the user this soon in the process.

In order to strengthen the development team's independence concerning external factors like other departments within the company, new approaches for research and user integration had to be developed. For example, to support early market research, different departments of the company were asked to participate in different user testing scenarios.

Research took five days in each sprint, but could easily be downsized to three days if all dependencies on other departments related to research had been non-existent.

Integrating non-specialist users from topically distant areas within the company made quick workshops and contests possible that allowed for early and swift user integration. This of course has to be taken as a small sample that in no way can be understood as representative, but rather as a point of inspiration for ideation.

Especially during the third sprint ideation was difficult to carry out without a final product in mind, since the project was already so far advanced. Many ideas were formed, that were not part of the final product but could be archived for future projects.

In order to strengthen the development team's independence concerning external factors like other departments within the company, new approaches for research and user integration had to be developed. It became clear, that the innovation department was lacking clear instructions on how to involve the user quickly and effectively. This included specific instructions rather than rough guidelines, if the innovation department was going to implement the methods in future projects. Defined outputs also help to underline a method's usefulness.

4 METHODS FOR QUICK, EASY AND VALUABLE USER INTEGRATION

The three selected methods described in this chapter centre around the key aspects time management and result reliance. All methods are designed to take up as little time as possible while yielding the most useful output possible. They were developed for the use case and were tested in the field while completing the development project.

All of the described user-centric methods that follow are based on observation and influence the users as little as possible by distracting them with time constraints or difficult tasks.

4.1 Method “pressure cooker”: uncovering existing habits with extreme time constraints

The method “pressure cooker”, as shown in figure 5, is designed to create pressure by giving the user a limited period of time to finish a task involving the product or service of interest. The time constraint is put in place to throw the users back into their existing habits and to prohibit any lengthy thought process while fulfilling a task. The user has no time to plan out the actions and has to simply act instead. For example, when developing new dishwasher baskets, this task might encompass loading the dishwasher with a set amount of tableware within a limited timeframe. This will consequently prohibit the user from analysing the space first, but instead create a sense of panic and trigger quick and rash decisions. The shorter and more unrealistic the time limit, the better.

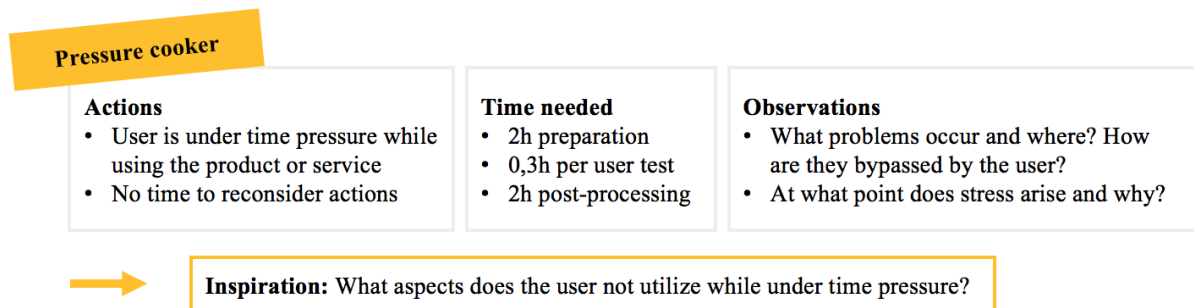


Figure 5. Method “pressure cooker”

This is exactly where the most valuable data is created. What problems arise in this situation are great input for later designs and can help to improve the product at hand. This method is therefore useful as an environment analysis of existing products as well as a prototype test for later development phases. Over the course of three days 24 users participated in this method. Data collection was made via photography and written observations.

4.2 Method “simultaneous interview”: provoking honesty through distraction

An additional approach to observe the user in a more unfiltered way is to combine simple tasks with an interview, as portrayed in figure 6. This way, the user is not able to focus on the task at hand while answering questions simultaneously. This method is by far the most difficult, since the interview has to be light enough in order to allow the user to complete a task without taking up too much attention and therefore turning into a full interview with no background assignment.

Simultaneous interview

| | | |
|--|---|---|
| Actions <ul style="list-style-type: none">• User is being interviewed while using the product/service• User falls back into old routines due to constant distraction | Time needed <ul style="list-style-type: none">• 0,5h preparation• 1h per user test• 0,5h post-processing | Observations <ul style="list-style-type: none">• Correlation and differences between answers and actions |
|--|---|---|



Inspiration: What is used constantly? What is used differently?

Figure 6. Method “simultaneous interview”

In contrast to the method “contextual inquiry” (Whiteside, Bennett and Holtzblatt,1988), the method “simultaneous interview” focuses on both the task and the interview, while the task at hand is the main source of inspiration rather than the interview, which is mostly meant to serve as a distraction.

The task could be something undemanding like loading the dishwasher at home, while answering simple questions about the product, in this case the dishwasher. The questions cannot be related to the current actions, since this would give the user the opportunity to overthink, but seem rather unrelated. This consequentially means that the interviewer has to assess the situation at all times and adjust the questions accordingly.

Since this method is more time consuming than the other two (including travel times), ten participants were chosen. The interviews were conducted at each user’s home and took around 30 to 60 minutes each. Photos were taken, but most observations were written in order to make the interview feel more private.

4.3 Method “brain twister”: revealing alternative uses through unsolvable tasks

“Brain twister” resembles the first method in the sense that it gives the user an unsolvable task concerning the product of interest. Instead of timing a certain set of exercises, this approach focuses on problem solving. Going back to the dishwasher example, this might entail a great amount of tableware, that has to be put in the dishwasher – but this time it is impossible to fit every single piece into it. Naturally this method is more time consuming than the timed version in 4.1, but it will also result in completely different observations. This time, the user is meant to puzzle over the task at hand and therefore focus on the task rather than the product.

Observations, as shown in figure 7, will reveal weak spots as well as opportunities, since the user is bound to find novel scenarios and solutions when faced with an impossible task.

Brain twister

| | | |
|---|---|--|
| Actions <ul style="list-style-type: none">• Give user a task to solve while using the product/service• Main focus is precision, not speed | Time needed <ul style="list-style-type: none">• 1h preparation• 0,5h per user test• 1h post-processing | Observations <ul style="list-style-type: none">• Complementing the method “pressure cooker“ to highlight differences• What alternative uses are found by the user? |
|---|---|--|



Inspiration: What aspects does the user utilize in a new way?

Figure 7. Method “brain twister”

23 users took part in this method. The 30-minute mark was met with 20 of the 23 users in contrast to method 1, because 30 minutes were the given limit and most users were keen to solve the problem. Many of them found very creative ways of loading all pieces into the dishwasher in order to “win” the challenge.

4.4 Feedback from participants within the use case

Participants of the use case are both users as well as product developers, both of which gave feedback concerning the methods and their application. 4.4.3 then translates the feedback into possible adaptations of the methods described.

4.4.1 Users

The most popular method among users was the pressure cooker. This was partly to its challenge-like design but also because it took only four minutes out of each user's day. This aspect has to be taken into consideration for the user to stay motivated during participation. The simultaneous interview was the most private method and took place at each user's home. Methods pressure cooker and brain twister took place at the company and with an available dishwasher at their lab, which meant that not all features were known to each user and that they had to get used to the model at hand first. Insights can also be used to determine how self-explanatory a product is, but will most definitely change the user's habits during participation, which needs to be taken into account.

4.4.2 Product developers

During all methods developers were integrated as users in order for them to a) understand the method and its application for future reference and b) gain insight in how the developers themselves use their own product. Even though they should have been experts, the developers did not use all features of their product under the given time constraints of the methods discussed.

It was important for the developers to know exact timetables of methods as well as their output and how to translate it into concepts. The latter was their greatest concern, since they were not equipped to evaluate the data that had been collected. It could be helpful to document what to look for and how to integrate it into the development process.

Product developers as well as the head of the department were surprised by outcome quantity and quality of the methods discussed. Being able to work independently from market research while ensuring guaranteed outcomes within a short time frame was greatly received and will be implemented in future projects.

4.4.3 Observations regarding further application of the methods

The developers' participation as users during data collection was vital to their acceptance of the methods and served as further input within the development process. The most prominent and usable results came from the method pressure cooker, which also took the least amount of time during preparation and evaluation. It was used to test an existing product, whereas method brain twister served as a verification of the first prototype. For future development projects, the methods need to be assigned to different product development stages in order to be implemented at the right time.

5 CONCLUSION

In today's accelerated product cycle (Kalogerakis 2010, Naccari 1997) and the market's growing need for innovation (Kalogerakis 2010), product development needs not only to become more agile but also more user focused in order to be sustainable. Approaches like Scrum and Design Thinking make user-centric and iterative development easier to obtain as shown in the use case, but still offer great potential for method development within their framework.

In summary, it can be noted that implementing aspects of Scrum within a hardware development project is not only possible but also greatly beneficial to its agility and shortens needed development time.

Adaptations to the development model need to be made in order to accommodate for hardware development, i.e. allowing for longer sprints during prototyping phases.

Furthermore, it was detailed how development teams need to be independent in prototype creation and user integration in order to stay agile and realise the proposed synthesis of Scrum and Design Thinking in hardware development.

To make the integration of the user as easy as possible, development teams need clear guidelines to follow that guarantee the desired output. Within the use case a variety of practical and precise user-centric methods of user integration were created, of which a selection was discussed in this paper. These methods can be adapted to each project's needs accordingly.

Upon completing the development project with the hardware manufacturer, their feedback was very clear: the methods developed would be making future projects that needed user integration a lot easier. What had been missing so far were precise instructions to follow rather than rough guidelines. This paper illustrates how user integration can be translated into clear procedures, which development teams can easily apply to their day-to-day routine. All described user-centric methods require little time in preparation, realisation and post-processing while producing predictable outcomes. The user-centric methods developed within the use case can be used for future application- and user-oriented projects in order to speed up the process and ensure that the product or service matches the users' needs and desires.

REFERENCES

- Brenner, W., Uebernickel, F. (2016), "Design Thinking as Mindset, Process, and Toolbox" in *Design Thinking for Innovation*. https://doi.org/10.1007/978-3-319-26100-3_1
- Cockton, G., Lárusdóttir, M., Gregory, P., Cajander, A. (2016), *Integrating User-Centred Design in Agile Development*. <https://doi.org/10.1007/978-3-319-32165-3>
- Doorley, S., Holcomb, S., Klebahn, P. (2018), *Design Thinking Bootleg*, d.school, Institute of Design at Stanford
- IDEO (2015), *The Field Guide to Human-Centered Design*, ISBN: 978-0-9914063-1-9
- Kalogerakis, K. (2010), *Innovative Analogien in der Praxis der Produktentwicklung*, Gabler. <https://doi.org/10.1007/978-3-8349-8674-0>
- Kusay-Merkle, U. (2018), *Agiles Projektmanagement im Berufsalltag*, Springer Gabler. <https://doi.org/10.1007/978-3-662-56800-2>
- Naccari, F. (1997), "Enterprise Integration in the User Industries — Needs and Current Solutions" in Kosanke, K., Nell, J.G. (eds), *Enterprise Engineering and Integration, Research Reports Esprit*, Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-60889-6_4
- Plattner, H., Meinel, C., Leifer, L. (2015), *Design Thinking Research – building innovators*. <https://doi.org/10.1007/978-3-319-06823-7>
- Ries, E. (2011), *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown Business, New York.
- Singh, M. (2008), U-SCRUM: "An agile methodology for promoting usability". In: *Proceedings of the AGILE 2008 Conference*, pp. 555–560. IEEE
- Sandmeier, P. (2008), *Customer Integration in Industrial Innovation Projects*, Gabler. https://doi.org/10.1007/978-3-8349-9746-3_5
- Schwaber, K. (1995), "Scrum development process". In: *Proceedings of the OOPSLA 1995 Conference, Languages and Applications - Workshop on Business Object Design and Implementation*
- Schwaber, K., Sutherland, J. (2017), *The Scrum Guide*, URL: www.scrum.org/resources/scrum-guide, accessed: 05.12.18
- Seely Brown, J. and Hagel, J. (2005), From push to pull: The next frontier of innovation, *The McKinsey Quarterly* No. 3, pp. 1–6.
- von Hippel, E. (1978), A customer-active paradigm for industrial product idea generation, *Research Policy* Vol. 7 No. 3, pp. 240–266. [https://doi.org/10.1016/0048-7333\(78\)90019-7](https://doi.org/10.1016/0048-7333(78)90019-7). https://doi.org/10.1007/978-1-349-03822-0_5
- von Hippel, E. (1978), Successful industrial products from customer ideas, *Journal of Marketing* Vol. 42 No. 1, pp. 39–49. <https://doi.org/10.1010.2307/1250327>.
- von Hippel, E. (1988), *The sources of innovation*, Oxford University Press, New York. Available at SSRN: <https://ssrn.com/abstract=2877276>. https://doi.org/10.1007/978-3-8349-9320-5_10
- Whiteside, J., Bennett, J., Holtzblatt, K. (1988), "Usability Engineering: Our Experience and Evolution", in *Handbook of Human Computer Interaction*, Helander, M. (Ed.). North Holland, New York, <https://doi.org/10.1023/A:1008171820407>