

THE CHALLENGES OF BECOMING AGILE – EXPERIENCES FROM NEW PRODUCT DEVELOPMENT IN INDUSTRY AND DESIGN EDUCATION

Nis OVESEN¹ and Chris DOWLEN²

¹Aalborg University

²London South Bank University

ABSTRACT

During the last decade *agile methods* have been a vast success in the domain of software development. This paper investigates whether these methods can be successfully transferred to the domain of physical product development in order to address the fundamental challenges of increased marked speed, development uncertainty and product complexity. The paper compares two cases from *industry* and *education* where agile methods are used in physical product development. The comparison between the cases is conducted within five thematic areas, which creates an overview of the challenges that may occur when implementing agile methods. The present paper is concluded by a discussion and conclusion drawing up the main challenges experienced and well as the benefits of utilising agile methods in physical product development.

Keywords: Agile methods, scrum, extreme programming, education, industry

1 INTRODUCTION

Heavy up-front planning represents a long management tradition in product development environments throughout the world. Gantt charts and Stage-Gate process models have become widely known industry standards and best practices taught in design and engineering education. However, the basic concept of these defined process control models – plan-your-work and then work-your-plan – often seems inappropriate when applied to development activities with uncertainty attached. This research project investigates the use of radically different and empirically based process control models originating from the domain of software development. Methods such as *Extreme Programming* [1] and *Scrum* [2] are part of the group of methods coined under the term *Agile Development* [3]. Earlier research has specifically mentioned these two agile methods as the most promising in theory [4]. Earlier contributions focusing on transforming agile methods are those of Highsmith & Cockburn [5] and Smith [6].

The paper is a result of an opportunistic international collaboration, bringing together experiences with agile methods from development departments in Danish companies and student mini-projects of MSc Design and Manufacturing Management students at London South Bank University in an evaluation of the methods' practical applicability to physical product design and development. The experiences are compared with the purpose of identifying common challenges as well as domain specific challenges. It is acknowledged that the use of two disconnected studies does not constitute a rigorous comparison but simply acts as a pointer towards the desirability of further studies taking place. Recommendations to students or industry take place not as a result of rigorous examination but pragmatism - what might be deemed to 'work' and provide results.

The challenges the methods pose are significant for product designers and developers in industry and education. If these methods might result in significant benefits, then they need to be evaluated by experienced designers in industry and by naïve student designers: these latter having few preconceived ideas of the management of development projects and thus providing a fair overview of utility compared with those in industry who disregard for untested methods. The educational setting provides an artificial environment, the findings of which can be communicated to industry.

The rest of the paper is composed as follows. The second section presents a brief overview of agile values and characteristics of agile methods. The third section presents the research setup and methodological approach. The fourth section presents five lenses for comparative analysis, which comprises the fifth section. Lastly, the sixth section discusses the general applicability of the methods to industry and design education, and the seventh section sums up the conclusions of the research efforts presented in the present paper.

2 CHARACTERISTICS OF AGILE METHODS

Agile Development as a term was established by a number of leading software pioneers in 2001. Together, they authored the *Agile Manifesto for Software Development* [7], which has gained vast success in the software industry throughout the last decade. The manifesto, consisting of a set of values plus 12 principles for best practice software development, promotes elements such as faster development cycles of only a few weeks, team-based responsibility, product specifications open for change, frequent prototyping and frequent process reflection. The most distinctive agile characteristics are briefly explained as a series of opposites to traditional development in Table 1 below:

Table 1. Traditional Development versus Agile Development

	Traditional Development	Agile Development
Management concept	Long phases of several months or years.	Short time boxes of few weeks in duration.
Initial specification	Static and specified to a high level of detail in the beginning of the project period.	Dynamic and typically described in broad terms in the beginning of the project period.
Product	Reviews with stakeholders in the end of each phase. Focus on verification.	Small iterations frequently inspected and changed midstream if market demands it.
Process	No formal process reflections in development team.	Frequent and formal team reflection and deriving corrective actions for increased development efficiency.

3 RESEARCH SETUP

This section briefly describes the research setup and data collection efforts of the two separate cases that this paper revolves around, namely A) student projects of MSc Design and Manufacturing Management students at London South Bank University and B) the development environments in Danish companies. In both cases, the research objective has been to investigate the applicability of agile methods in physical product development. However, due to their different nature, the process of collecting the data has been somewhat diverse. An overview of the research setup is presented in Figure 1 below. The separate approaches are outlined in the subsequent sub sections.

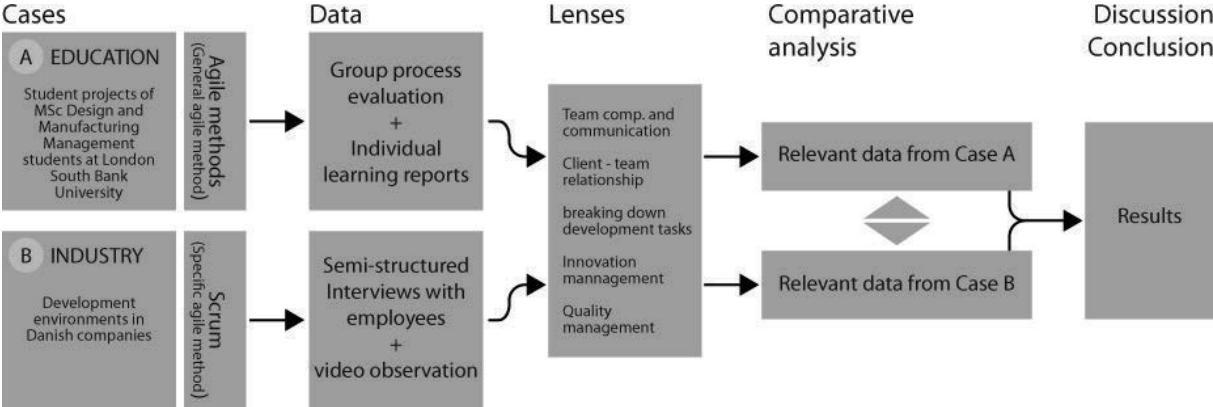


Figure 1. Overview of research setup

3.1 The Education Case

The process of student evaluation of the agile methods was to set a relatively small group of students on the MSc in Design and Manufacturing Management a design task that also included an evaluation of any of four of the agile processes: Scrum, Feature-Driven Development, Extreme Programming and Pragmatic Programming. The relatively simple design task presented to the students was to design a small deodorant container suitable for use in hotel rooms as part of a toiletries set. Students worked in three groups of four or five: project team selection was left to the students, although it was assumed that the teams remained constant. They were left to organise the teams as they wished.

Student output was split between the design work undertaken and a critique of the agile methods. Presentation of the results was in a group lecture format, backed up by a report. Each student also had to produce an individual reflective report. Students assessed each other in the groups.

The students only had a short time to produce their work and thus had to have relatively short timescales for their scrum periods: they had these more frequently than their classes and performed significant iterations of the design. Two groups made the decision to concentrate on the developmental cycle and one on gathering and developing the customer requirements. One of the groups produced ten different product concepts: the one focusing on customer requirements had four different iterations of the brief, developed through discussion with a (real) hotel manager as customer, and this resulted in an evolution of the holistic concept for the product and provided effective customer feedback throughout the development process.

Table 2. The fourth iteration of one group's customer requirements

INDUSTRY	TASK LIST
If customer feel to take the product	Not removable
Not to touch the contents of product	Pushable by button
Easy to reach	Display in two sections

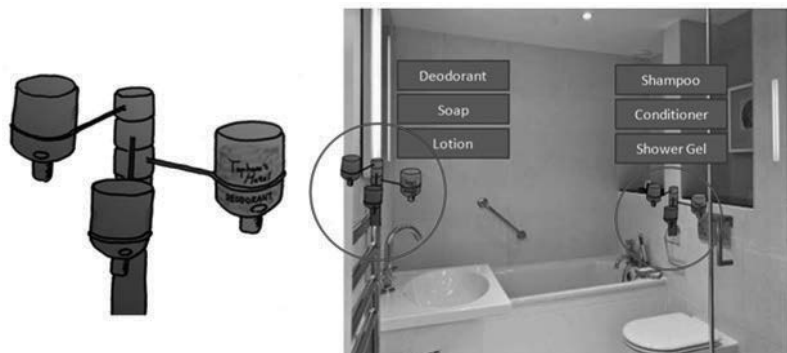


Figure 2: The fourth iteration of the group's overall product concept

All students claimed to have learnt a significant amount through the exercise, and most had enjoyed the time: they felt they had an adequate chance to appraise the agile methods and to identify advantages and disadvantages associated with the methods they had used.

3.2 The Industry Case

The six Danish companies involved in this comparative analysis were all in the process of implementing agile processes in their respective development environments, more specifically the Scrum method. Their experiences ranged from only a few months with the method to three years and all companies were operating on a global market. Information about their interpretation of Scrum and the challenges experienced when implementing the method were collected through video observation and series of semi-structured interviews with employees in each company. The data from these research efforts therefore has character of quotations, thematically sorted to shed light on a series of different aspects and experienced challenges.

4 LENSES FOR THE COMPARATIVE ANALYSIS

The overall theme for the analysis is *the challenges of utilising agile methods from software in product development activities*. This section briefly presents the lenses through which the comparison between case A and case B is carried out.

4.1 Team composition and communication issues

The concept of High Performance Teams is an important issue in agile software development as some of the general characteristics in agile methods are self-organising teams and large development responsibility placed directly on the team. This lens focuses on challenging issues in regards to how teams are composed and how communication flows within the team.

4.2 Client – Development team relationship

In the Agile Manifesto “customer collaboration” is preferred over “contract negotiation”. In most agile methods this is reached through close contact and frequent inclusion of the customer to the development process. However, the client to a development team can be rather ambiguous compared to pure software development projects. This lens compares the client relationship between the cases.

4.3 Breaking down development into short cycles

In agile methods, the development is conducted through short time-boxed cycles where development tasks are broken down to small quantifiable work packages of only a few hours of work. This lens focuses on the implications, which the emphasis on full transparency and resource estimation may have on the development activities.

4.4 Innovation management

Innovation management has become an important measure of success and a competitive advantage in the industry if mastered well. Quite a few of the agile methods are relatively heavy in directions and guidelines when it comes to how to manage the development activities, but how they facilitate and foster creativity and innovation efforts, both internally and externally, is articulated less clearly. It is the focus of this lens.

4.5 Quality management

The last focal point in the comparative analysis is *quality management*. How is quality management handled in physical product development assisted by agile methods? Do methods like Scrum and Extreme Programming provide any directions for quality management?

5 COMPARATIVE ANALYSIS

This section consists of the comparative analysis between case A from the educational domain and case B from the industrial domain. The analysis is found in the following sub sections and is organised according to the five lenses presented in the section above.

5.1 Comparison: Team composition and communication issues

Due to the high level of cross-functionality compared with software projects, development teams from the industry experienced difficulties in acting as closely as they initially wanted. The Scrum framework claims team members should be able to take over each other’s work as all tasks are team responsibilities, but this proved difficult in practice – because the differences in competencies between developers were too large. One industrial developer commented: “*At the Daily Scrum meeting it can be a challenge that someone doesn’t understand why they have to listen to what everyone else in the team are doing, when they are that specialised into different areas as they are.*”

However, agile methods emphasised close team coordination through frequent short formalised meetings generally proved to be highly valuable despite the large span in professional competencies. Legacy power structures exist in all companies and create unwanted and conflicting hierarchies in the development environment. These structures interfere with lightweight decision-making processes, which are an important part of the agile development and hamper efficient development rooted in the self-organising team.

With a short project timeframe and absence of legacy power structures, all students reacted positively to team composition and frequent team meetings. The groups were devoid of hierarchy and thus had

flat management structures. Though students had different educational backgrounds, they found agile principles beneficial for teamwork. One student claimed: *“Meeting sessions made the real difference and we always had a further direction at the end of every meeting session”*. One student felt teams should have been selected, and that they might have benefited from allocating members specific tasks.

5.2 Comparison: Client – Development team relationship

Software frequently works on a contract basis: not always so for products. In industry, the client is typically an internal marketing representative. In isolation, this may not entail specific challenges to the development team, but with an internal client agile contracting practices are rarely accepted. The consultation with an internal client can create potential risk of alienating the end user from the development team, and often requires acceptance from several stakeholders to allow end users into the development environment.

In education, the client-team relationship was relatively problem free - but is contrived and unnatural. The students were set up with an imaginary client manufacturing small toiletries for hotels and similar establishments. A student reflects upon the group’s collaboration with a real customer: *“In early stage, our customer (...) had a very negative opinion that this project could not be brought into being, however, after experiencing design improvements and changes using agile method, and seeing his requirements realized, he gradually changed his attitude positively.”* They benefitted from being independent and removed from agendas forced by an organisation or higher-level stakeholders.

5.3 Comparison: Breaking down development into short cycles

Students felt that whilst the project was a relatively simple design project, it progressed significantly faster than without the use of agile methods. Frequent iterations took place both in customer requirements and developmental processes and team buy-in was strong, enabling effective design thinking to take place.

In industry feelings were mixed. Whereas most of the interviewed employees in the six companies experienced higher efficiency in the development activities due to the frequent progress reviews, the short iterative time boxes added frustration to some team members. A developer expressed this: *“The biggest challenge, that I have noticed, is definitely the breakdown of tasks to deliverables that can be fitted in to two or four week Sprints. It is a change in attitude rather than a technical challenge”*. This is a recurrent challenge in all the industrial development environments of case B.

5.4 Comparison: Innovation management

Students had previously been provided with toolkits of development methods. One shortcoming of agile methods, they felt, was the lack of methods for assessment of design solutions and the unstructured results from group decision-making meetings. They commented that the methods would not really be suitable for more complex design problems or where there were significant systems and specialist issues. They felt they were concentrating on holistic and generalist issues of the product – although they had previously designed injection moulded components they assumed they would choose. They did not comment on the concrete nature of their outcome compared to the virtual nature of the software engineering problem.

There was a tendency that design efforts were not well integrated into the iterative cycles in industry. All involved companies used Scrum, which does not provide any specific guidance for managing innovation and design efforts. On the other hand, most agile methods enable development teams to open up for new innovation midstream. After all, one of the agile principles is “embrace change”. They just do not provide specific guidance for fulfilling this principle.

5.5 Comparison: Quality management

Agile methods do not provide processes that guarantee quality control. Students felt they wanted to use other methods for doing this (they were not prevented from doing so) and thought the qualitative, human discussion processes of scrum needed to be augmented by quantitative evaluation such as a matrix process.

In industry, all companies followed traditional stage-gate models with well-established procedures for risk analysis and quality management. The agile method of Scrum did not contribute to this. Instead Scrum was in general seen as an addition to their legacy of existing management systems and product development and launch processes.

6 DISCUSSION

The student groups gave significant benefits to using agile methods: they commented on the high customer involvement, reduced failure costs, prompt feedback and effective communication. In terms of keeping a project to time, they agreed that agile processes allowed quick starts, were quick to allow adaptations and modifications to take place and be agreed without having to go through swathes of documentation. The personal involvement aspects were commented on significantly: group meetings were positive and purposeful: more than one student commented on how they had enjoyed the processes.

On the downside, the students felt there were significant limitations for physical products. Agile, by its nature, appears to be limited to simple products aimed at a small number of favourable customers. Students were concerned about the lack of inherent quality tools and felt the subjective nature of decision-making was negative.

As in education, industry contained both challenges and benefits of implementing agile methods. The challenges included complications of integrating agile values and procedures in a traditional development environment, such as concern at stakeholder level and antibodies in the development teams. Practical challenges of breaking down development tasks or maintaining close communication through daily meetings were also of significant drawbacks. On the other hand, benefits like faster development and focus on efficiency and team transparency were highly positive additions to their existing development culture.

7 CONCLUSION

Does agile work for products? That is probably not the right question to ask. There are not two separate, analogous processes. It is more useful to ask in what ways agile design methods might be incorporated into product design. Certainly the emphasis on developing customer requirements is perceived as positive, as is their effect on project development time. In order to gain the best from the methods, they probably need to be used in conjunction with other methods such as QFD and evaluation and selection matrices. But they have a future and should not be ignored.

It is clear that development environments in industry have significant challenges of implementing and fully benefitting from the agile methods. However, it is also clear that agility is gaining increasing interest from the product development industry, which will eventually transform agile values and methods into truly useful additions to their existing processes, and students should arguably be prepared for this. Perhaps the most telling comment was from one student who commented "*I think Agile is definitely not just for team projects and can be applied to personal projects. Therefore, this Agile assignment can be changed from group-based project to individual one in order to assess individual performance.*"

In this particular instance there was one student whose comment when working with the hotel manager was, "*it made it such a fun to see how much change has happen*". It could be that the agile manifesto, focusing on human aspects, has given the students a human-based approach and outlook that may be lacking with more traditional methodologies.

REFERENCES

- [1] Beck K. and Andres C. *Extreme Programming Explained: Embrace Change*, 2nd ed. 2004 (Addison-Wesley Professional, Boston USA)
- [2] Schwaber K. and Beedle M. *Agile Development with Scrum*, 2002 (Pearson Education, New Jersey USA)
- [3] Highsmith J. et al *Manifesto for Agile Software Development*, agilemanifesto.org, 21-12-2010
- [4] Oversen N., Eriksen K. and Tollestrup C. Agile attitude: Review of Agile Methods for use in Design educations. In *The 13th International Conference on Engineering and Product Design Education, E&PDE2011*, City University, London, UK, September 2011, pp.505-510 (Institution of Engineering Designers & The Design Society, Glasgow, UK)
- [5] Highsmith J. and Cockburn A. Agile Software Development: The Business of Innovation. *Computer*. 2001, 34(9)
- [6] Smith P.G. *Flexible Product Development: Building Agility for Changing Markets*. 2007 (Jossey-Bass/Wiley, San Francisco)
- [7] Highsmith J. et al. *Manifesto for Agile Software Development*, agilemanifesto.org, 28-02-2012