This paper was presented at The ISPIM Innovation Conference – Innovating Our Common Future, Berlin, Germany on 20–23 June 2021.
Event Proceedings: LUT Scientific and Expertise Publications: ISBN 978-952-335-467-8

EDiT – Requirements of Enabling Distributed Collaboration in Product Development Teams

Katharina Duehr*

Karlsruhe Institut of Technology (KIT), Kaiserstr. 10, 76131 Karlsruhe, Germany.
E-mail: katharina.duehr@kit.edu

Amelie Schiele

Karlsruhe Institut of Technology (KIT), Kaiserstr. 10, 76131 Karlsruhe, Germany.
E-mail: amelie.schiele@student.kit.edu

Hannes Mueller

Karlsruhe Institut of Technology (KIT), Kaiserstr. 10, 76131 Karlsruhe, Germany.
E-mail: hannes.mueller@student.kit.edu

David Kopp

Karlsruhe Institut of Technology (KIT), Kaiserstr. 10, 76131 Karlsruhe, Germany.
E-mail: david.kopp@student.kit.edu

Manfred Hofelich

TRUMPF Werkzeugmaschinen GmbH + Co. KG, Johann Maus-Strasse 2, 71254 Ditzingen, Germany.
E-mail: manfred.hofelich@trumpf.com

Albert Albers

Karlsruhe Institut of Technology (KIT), Kaiserstr. 10, 76131 Karlsruhe, Germany.
E-Mail: albert.albers@kit.edu
* Corresponding author

Abstract: Not only caused by the Covid-19 pandemic but also by the ever increasing dynamic interconnectivity of advanced systems, teams work more and more from different locations. Distributed collaboration poses new, unexpected challenges to the teams. To avoid efficiency and effectiveness losses in the product development process, teams need to continuously improve their distributed collaboration. Targeted methodical support can help to identify improvement potentials and ultimately counteract the efficiency and effectiveness. Therefore, the focus of this contribution is to identify requirements for a method that enables development teams to continuously improve their distributed collaboration. Finally, the requirements will be operationalized to build the foundation for a method that is accepted by the user, applicable, and that creates a direct benefit

Keywords: Distributed teams; collaborative innovation; new work; method development; requirements; Delphi study; product development; user-centered, improving distributed collaboration.

1 Introduction

In response to emerging changes caused by the ever increasing dynamic interconnectivity of advanced systems, more and more product development teams are collaborating across distributed locations (Dumitrescu et al., 2021). This tendency is further propelled by the ongoing Covid-19 pandemic, often leaving teams no other option than working from different locations. However, transitioning to distributed collaboration, development teams face new, unexpected challenges, which will have to be solved continuously. Not addressing those sufficiently can leave team members feeling overwhelmed and left alone in the new working environment resulting in losses concerning the efficiency and effectiveness of distributed product development processes (Duehr et al., 2020). Targeted methodical support can help to identify improvement potentials and ultimately reduce the efficiency and effectiveness losses just mentioned (Ehrlenspiel & Meerkamm, 2013).

Although numerous efforts have been made in recent years to provide methods for improving product development processes, only a few have been presented to support a continuous improvement process for distributed collaboration (Duehr, Efremov et al., 2021). Since product developers, as users of methods, form the center of product development, it is crucial to base the design of such a method on their needs (Albers et al., 2019). Furthermore, methods should enable adaptability to an individual development situation (Birkhofer et al., 2005). They must be accepted by the user, applicable as well as create a direct benefit (Blessing & Chakrabarti, 2009; Marxen, 2014). Currently, however, sufficient knowledge does not exist about the requirements asked for by the user regarding a method that enables product development teams to continuously improve distributed collaboration. Therefore, the focus of this contribution is to identify requirements for such a method to enable continuous improvement in distributed development teams. The operationalization of these requirements is supposed to reflect the context-specific development situation.

2 Current understanding

Distributed product development teams

Dörner (1976) describes a problem as a deviation between the undesirable actual and the desired target state. The path between those two states is only partially known (Albers & Braun, 2011; Dorst, 2006). Product development is understood as a problem-solving process with different problems centered in a basic challenge and entails all activities required for product development (Albers & Braun, 2011; Dorst, 2006). These processes and workflows must be coordinated to manage capacity and economic challenges (Schuh, 2012). According to Kirchner (2020), product development is not only about information processing but is a way of finding a solution through multiple influences and requirements. In the process of product development, task clarification takes place through product definition, idea generation, concept development and designing the product (Schmidt, 2017). These phases take place at the levels of organization, product and process (Bullinger et al., 1997). The boundaries between the three levels are fluent and can be described as such: A product is created in a process that is embedded in an organizational structure where product development is part of a larger system as e.g. the company and is linked to other business processes (Gierhardt, 2002).

Fundamental to this perspective is an integrated or system-oriented view of the products and the product development process. The so-called ‘systems engineering’ describes development tasks or challenges during the development of a system (Dumitrescu et al., 2021). Within the context of systems engineering, products and processes are modeled under the assumption of interdisciplinary and the development tasks are enabled from different perspectives. This supports more consideration of alternative approaches within the product development process (Schömann, 2012). Especially for the development of increasingly complex products, which often require the combination of knowledge from the fields of mechanical engineering, electrical engineering or information science, the collaboration of experts from various interdisciplinary domains is necessary (Bavendiek et al., 2018).

Increasing interdisciplinarity of product development teams to master technical and organizational complexities in product development, demands for collaboration of developers working from different locations being connected with information and communication tools (Gierhardt, 2002). Early on, Welp (1996) described the goal of distributed product development as to manage development processes across multiple locations through internal or external partnerships, cooperation or alliances, to optimize time, quality and cost. Distributed product development can be described as collaborative processing of different subtasks of product development while focusing on the aspects of cooperation, coordination and communication (Krause, 1998). Considering the process elements involved, this is the combination of people, material, activities, methods and tools for the effective development of products across locations (Gierhardt, 2002). Apart from the term ‘distributed teams’, the term ‘virtual team’ is frequently used. A virtual team is understood as a group of people who work independently across space, time and organizational boundaries on a common goal with the help of technology (Lipnack & Stamps, 1997). This definition not only emphasizes the technological aspect as a key feature but also expands the geographical dimension to include a time and organizational component.

Key challenges for distributed teams

The most obvious challenges in distributed teams arise from the physical distance between team members and express themselves particularly in communication difficulties (Ahuja, 2017). Increased waiting times among team members in different time zones and less informal, spontaneous communication in distributed teams can lead to a loss of information and slowed down exchange (Herbsleb et al., 2000; Larsson et al., 2003). Different languages and cultural backgrounds can cause misunderstandings and generate conflict potentials that stand in the way of a common understanding of goals (Herbsleb et al., 2000).

One of the most visible differences in distributed communication compared to co-located settings is the use of technical aids (Stöger & Thomas, 2007). To substitute personal meetings, telephone and video conferencing tools are used, saving travel time and money (Stöger & Thomas, 2007). However, Kuster et al. (2011) point out that media-based communication can rarely ensure truly complete communication. Compared to personal conversations body language, eye contact, posture and extra-communicative actions are often partially or completely lost, depending on the medium (Kuster et al., 2011). Electronic media are not only used for communication purposes but also play a role in the development of specific tasks (Konradt & Hertel, 2002) requiring additional competencies from developers (Bavendiek et al., 2018).

Beyond these mostly technology- and communication-related challenges, Gaul (2001) additionally names organizational complications as a key challenge regarding distributed product development. These include a possible outflow of know-how to partners as well as mutual dependency and data security risks. Due to the low level of experience with distributed product development in many teams, there is often a lack of organizational approaches to achieve the desired level of formalization and alignment on the actual development situation of distributed processes (Duehr et al., 2019). This results in a high expenditure of time for coordination activities, which leads to a loss of working time for processing actual development tasks (Bavendiek et al., 2017).

Dillenbourg et al (1996) state: “collaboration is in itself neither efficient nor inefficient.” He argues that it is the task of research to determine the circumstances under which collaboration is efficiently possible. Accordingly, the individual boundary conditions of the company and the development task, i.e. the system of objectives, must be determined to subsequently adapt the operation system as a socio-technical system, including activities, methods and tools as well as the resources needed for all development activities (Albers et al., 2016). Methods are one way to support distributed teams in helping to deal with these challenges in a structured way and have already proven their value in practice (Lindemann, 2016). However, it is crucial to address the specific needs related to boundary conditions of a development situation to continuously master the individual challenges (Duehr et al., 2019). Only in this way can the methods used in distributed product development employ their full potential (Birkhofer et al., 2005).

A method to support distributed product development teams

In order to characterize individual development situations, Albers et al. (2020) described influencing factors relevant for successful distributed product development that can be clustered in fields of action of distributed product development. According to Gericke et al. (2013) the sum of influencing factors, also described as context factors, represent all factors that influence the application of design projects, processes and methods. Therefore, in this contribution, influencing factors are interpreted as ‘levers’ for implementing measures in the development environment to improve distributed collaboration Albers et al. (2020) assigned the success-relevant influencing factors of distributed product development to the design dimensions of ‘technology’, ‘organization’ and ‘people’.

The design dimension of ‘people’ describes all interpersonal and group dynamic processes but also the capabilities of individuals that are relevant for distributed collaboration. In addition to the capabilities of individuals, the ability of different characters to work together in a team is a particular challenge (Albers et al., 2020). The design dimension of ‘technology’ includes the design of collaboration through methods and tools. ‘Organization’ brings together the two other fields of action and includes task, structure and process organization to support collaboration. In this regard, the integration of different departments is of particular importance. This means that careful organization is required in the coordination of various players, as there is a mutual dependency (Duehr et al., 2020).

As was pointed out, distributed product development is an extremely complex undertaking that poses major challenges for many teams. Both literature and empirical findings indicate that practitioners are seeking methodological support to improve distributed product development in a structured way (Duehr, Kavakli, & Albers, 2021). Although numerous efforts have been made recently to provide methods for product development in general, there is still little support for the identification and exploitation of improvement potentials in distributed development teams.

A method can support the identification of improvement potentials by including success-relevant influencing factors represented in the fields of action of distributed product development (Albers et al., 2020). One initial approach that attributes the influencing factors to the three dimensions of the TOP model to support the improvement process holistically was presented by Duehr et al. (2020). The following objectives were stated that also represent the basis for the initial process steps of the method under development (Duehr et al., 2020):

- Consideration of the prevailing development situation
- Identification of individual improvement potentials in communication processes of distributed product development
- Definition of measures to address the identified improvement potentials

Although a method should support the improvement of distributed collaboration, useful outcomes cannot be guaranteed. A fit must exist between the need for support encountered in the development team and the value offered by a proposed method (Gericke et al., 2017). Accordingly, the question has to be asked which requirements have to be met by such a method to increase the chances of being successful.

Evaluation principles for methods in product development

From an academic perspective, successful method development is distinguished by the fact that it finds its way into practice (Gericke et al., 2017; Marxen, 2014). Although methods are an important part of product development and extensive research activities revolve around them, the adoption in the development practice is often very slow (Gericke et al., 2017). According to Jänsch (2007) transfer problems are not simply acceptance problems but also application, teaching, presentation and documentation problems. Tangible results, a clear understanding and ease of application are key requirements for successful method development (Gericke et al., 2017).

To increase the probability of a successful transfer into practice and to ensure that a developed method support is tailored to the specific needs of its users, (measurable) success criteria must be defined as part of an iterative method development process (Blessing & Chakrabarti, 2009). These serve to clarify the added value for practice and are made tangible by the definition of measurable requirements (e.g. functionality, usability, costs or life cycle). According to Albers et al. (2013), requirements should be derived from goals and other boundary conditions. Vice versa, a requirement is the set of conditions or properties that is needed to achieve the goal (Pohl, 2007). Requirements can be both qualitative and quantitative data and can be observed by the researcher (Blessing & Chakrabarti, 2009).

Evaluation is needed to determine whether the application of the proposed support indeed leads to a fulfillment of the determined requirements (Blessing & Chakrabarti, 2009). The major goal of evaluating methods is to relate the generated outcome to the identified criteria. In the context of the Design Research Methodology (DRM), a distinction can be made between three types of evaluation: 'success evaluation', 'support evaluation' and 'application evaluation' (Blessing & Chakrabarti, 2009).

On the highest level, the success evaluation is used to measure the added value in terms of meeting the formulated objectives. This is the most comprehensive evaluation type, which can only be carried out if the applicability of the support is guaranteed by the application evaluation (Blessing & Chakrabarti, 2009). The emphasis of this evaluation type is not on whether the support meets the requirements, but on how it does so. The key question at this level is: Does the application of the support mechanism have the desired effect on the measurable requirements?

The support evaluation deals with the examination of the developed mechanism regarding its functionality and consistency. It should be performed during the whole development of the method starting from the very beginning. The overarching question of interest here is: Does the method basically work as desired? Applied to the setting of distributed product development, the question to be answered is whether the developed method can meet the previously identified need for support (Blessing & Chakrabarti, 2009).

Badke-Schaub et al. (2011) mention the insufficient usability or inadequate presentation of methods as a reason for deficits. This is where the application evaluation comes into play. The main question in this regard is how easily the method can be handled by the user. For example, this evaluation type examines whether a method is intuitively structured or has the necessary level of detail (Blessing & Chakrabarti, 2009).

3 Research objective and methodology

The state of research indicates that the increasing dynamic interconnectivity of advanced systems demands the collaboration of development teams across different domains (Dumitrescu et al., 2021). This collaboration often takes place across many locations and poses challenges for the development teams. Therefore, the overall goal is to enable the development teams to continuously improve their distributed collaboration. For this, a method has to be developed. The goal of this contribution is to identify requirements of a method that enables continuous improvement in the collaboration of distributed development teams. In addition, measures will be derived of how these requirements can be operationalized in the subsequent development of the method. Thus, the following research questions will be answered:

RQ1: What are the requirements for a method to enable development teams to continuously improve distributed collaboration?

RQ2: How can the requirements be interpreted and operationalized as measures, elements, and activities in the method development?

A two-stage expert study based on a Delphi (Häder & Häder, 2000) study is conducted to derive requirements of a method to enable development teams to continuously improve distributed collaboration. The source for the determination of requirements in the first Delphi stage was a series of expert interviews as well as literature research based on the fundamental objectives of the method. For the qualitative determination of requirements in the first stage of the Delphi study, five semi-structured interviews with experts from different business divisions and with expertise (> 3 years) in distributed product development are conducted. The experts covered the following positions: Scientific Manager, Innovation Manager, Interim Manager in the field of improving distributed collaboration of parent company and subsidiary, Managing Director, Master Student Mechanical Engineering. The semi-structured interviews lasted between 60 and 90 minutes to ensure a deep exploration of the research topic. Subsequently, the results are consolidated and transferred into requirements. For this purpose, statements about possible requirements for the method to be developed were extracted from the transcripts and then summarized with similar statements. Since the method is intended to provide holistic support for improving distributed collaboration and subsequent validation of the method, the requirements are assigned to the evaluation criteria based on the Design Research Methodology (DRM) by Blessing and Chakrabarti (2009). These are the *support performance*, the *applicability* and the *contribution to success.* To evaluate the relevance of the identified requirements in the second Delphi stage, the requirements were transferred to an online questionnaire. A five-point ordinal scale was used to assess the relevance.

In total, 125 participants completed the questionnaire. Table 1 shows the distribution of survey participants across their fields of activity. 20 % of the respondents have been working for their company for less than one year. Between 1 and 3 years, approximately 13 % have been working for their company, again 24 % for 4 to 5 years, 19 % for 6 to 10 years and 22 % for more than 15 years. 74 % of the survey participants fully or rather agreed with the statement that they have experience in distributed product development.

Table 1 Affiliation of survey participants by field of activity (n = 125)

Field of activity	SHARE OF PARTICIPANTS [%]
Student	33
Science and research	12
Mechanical and plant engineering	9
Automobile manufacturer	9
Other vehicle manufacturing	1
Automotive supplier	2
Aerospace	13
Electronics / Electrical engineering	7
Information technology	4
Other	10

To make statements about the relevance of individual requirements, various stochastic methods were applied. A Friedman test was conducted to compare the relevance ratings of the different requirements. Therefore, the individual ranking of the requirements for each of the three DRM evaluation types was calculated for every participant based on their respective relevance rating. Subsequently, the rank sum was calculated for the individual requirements. The comparison of the rank sums using the Friedman test revealed a significant difference between the associated requirements for each evaluation type. The chi-square independence test was used to show the stochastic independence of the characteristics. For a differentiated consideration of the relevance differences of the individual requirements of an assessment type, pairwise post-hoc tests were then performed. In order to neutralize the alpha error accumulation due to the multiple comparisons, a Bonferroni correction of the p-values was performed. A significant difference in the relevance score of two requirements was assumed in the case of a corrected p-value smaller than 0.05. Based on the results of the post-hoc tests, the requirements were finally grouped into rank groups. Rank groups were formed in such a way that the mean ranks of the requirements within a rank group did not differ significantly from each other. In addition, the same statements on the significance of the differences in a rank group compared to other rank groups always apply to all requirements in a rank group. Due to the homogeneous high relevance of all requirements, the requirements were all focused and operationalized as measures, elements and activities to finally develop the EDiT method (Enabling Distributed Teams) based on the understanding of the existing theory and the requirements profile derived from the presented study.

4 Findings

Requirements for the method

In total, 18 requirements for the method under investigation were identified in the two-stage Delphi survey and evaluated according to their relevance.

Success evaluation

The six identified requirements of the method for the success evaluation according to the DRM are shown in Table 2. According to the results of the relevance evaluation in the course of the second Delphi stage, the requirements can be classified into two rank groups (cf. Table 3). While the basic objective and the need for a method have already been described in previous studies, it can be further specified based on the identified requirements for the method's contribution to success. Accordingly, the requirement for the successful contribution of product development teams in the continuous improvement of distributed collaboration is first the support in the improvement of distributed collaboration of product development teams (E1). Furthermore, the effort-benefit ratio of the process to improve distributed collaboration should be positively influenced (E2) and the efficiency (E3) and effectiveness (E4) of distributed product development teams should be improved.

Table 2 Requirements for the success evaluation of the method

	The method should …
E1	… support the improvement of distributed collaboration within product development teams.
E2	… positively influence the effort-benefit ratio of the process for improving distributed collaboration.
E3	… improve the efficiency of the distributed product development team.
E4	… improve the effectiveness of the distributed product development team.

The Friedman test based on the evaluation results of the second Delphi round revealed significant differences in the relevance of the different requirements to the contribution to success with a test statistic of $\chi 2(3) = 26.726$ and a p-value of $p < 0.001$. From the results of the pairwise comparisons with the post-hoc test in Table 3, it appears that the method is intended to improve the distributed collaboration of product development teams. The corresponding requirement E1 forms its own rank group with significantly higher relevance than the requirements of rank group 2.

Table 3 Results of the relevance evaluation of the requirements for the success evaluation of the method (n = 125)

Rank Group	*Requirement*	*Medium Rank*	*Relevance**	*E1*	*E4*	*E2*	*E3*
1	E1	2,11	1,6		0,053	● 0,007	● 0,001
2	E4	2,54	1,9	0,053		1,000	1,000
	E2	2,64	1,9	○ 0,007	1,000		1,000
	E3	2,72	1,8	○ 0,001	1,000	1,000	

● Requirement (row) significantly more relevant than requirement (column)
○ Requirement (row) significantly less relevant than requirement (column)
0,1 corrected p-value from pairwise post-hoc test
*Relevance of the requirement [1(essential) - 5(very unimportant)]

Support evaluation

Table 4 shows the six requirements for the support evaluation according to the DRM identified during the expert interviews. The prerequisite for consistent support in distributed product development is to enable support in understanding the influencing factors (U1). Furthermore, the identification of critical activities (U2), the analysis of improvement potentials of collaboration (U3), the definition of measures to develop improvement potentials of collaboration (U4), the implementation (U5) and evaluation of defined measures to improve collaboration (U6) of distributed product development teams should be supported.

Table 4 Requirements for the support evaluation of the method

		The method should …
U1	…	support the understanding about the factors influencing distributed product development.
U2	…	support the identification of critical activities of distributed product development.
U3	…	support the analysis of improvement potentials of the collaboration of distributed product development teams.
U4	…	support the definition of measures for the development of improvement potentials of the collaboration of distributed product development teams.
U5	…	support the implementation of defined measures to improve collaboration among distributed product development teams.
U6	…	support the evaluation of implemented measures to improve the collaboration of distributed product development teams.

The Friedman test revealed significant differences in the relevance of the different support performance requirements with a test statistic of $\chi 2(5) = 43.696$ and a p-value of $p < 0.001$. From the results of the pairwise comparisons with the post-hoc test in Table 5, it is clear that the method is intended to support the identification of critical activities in distributed product development. The corresponding requirement U2 forms its own rank group with significantly higher relevance and correspondingly lowest mean rank compared to the requirements of rank groups 2 and 3. The differences in the mean relevance scores of the two requirements with the next lowest mean rank are not significant, so they form the joint rank group 2. All requirements of rank group 2 are significantly more relevant than those of rank group 3.

Table 5 Results of the relevance evaluation of the requirements for the support evaluation of the method (n = 125)

Rank Group	*Requirement*	*Medium Rank*	*Relevance**	*A1*	*A2*	*A3*	*A6*	*A5*	*A4*
1	A1	2,75	2,4		1,000	● 0,003	● 0,002	● 0,000	● 0,000
2	A2	2,79	2	1,000		0,005	● 0,004	● 0,000	● 0,000
3	A3	3,64	2,3	○ 0,003	0,005		1,000	1,000	● 0,039
	A6	3,65	2,4	○ 0,002	○ 0,004	1,000		1,000	● 0,046
4	A5	3,82	2,2	○ 0,000	○ 0,000	1,000	1,000		0,402
5	A4	4,35	2,6	○ 0,000	○ 0,000	○ 0,039	○ 0,046	0,402	

● Requirement (row) significantly more relevant than requirement (column)
○ Requirement (row) significantly less relevant than requirement (column)
0,1 corrected p-value from pairwise post-hoc test
*Relevance of the requirement [1(essential) - 5(very unimportant)]

Application evaluation

Six applicability requirements were identified for the application evaluation after the DRM, which are shown in Table 6. For the applicability of the requirements in distributed product development, the method should have an appropriate effort-benefit ratio (A1). Furthermore, it should be easy to use for the development team (A2), be structured in meaningful work steps (A3), have an appropriate level of detail (A4), be able to be integrated into existing processes (A5) and be applicable in different development teams (A6).

Table 6 Requirements for the application evaluation of the method

		The method should …
A1	…	have a reasonable ratio of effort and benefit.
A2	…	be easy to use for the development team.
A3	…	be divided into meaningful steps.
A4	…	have an appropriate level of detail.
A5	…	be able to be integrated into existing processes.
A6	…	be applicable in different development teams.

The Friedman test revealed significant differences in the relevance of the different applicability requirements with a test statistic of $\chi2(5) = 91.407$ and a p-value of $p < 0.001$. According to the results of the relevance assessment, the requirements can be classified into five rank groups (Table 7). The requirement with the highest relevance and correspondingly lowest mean rank is the requirement for an appropriate cost-benefit ratio (A1). It forms a separate rank group with significantly higher relevance than the requirements of the other four rank groups.

This paper was presented at The ISPIM Innovation Conference – Innovating Our Common Future, Berlin, Germany on 20-23 June 2021.
Event Proceedings: LUT Scientific and Expertise Publications: ISBN 978-952-335-467-8

Table 7 Results of the relevance evaluation of the requirements for the application evaluation of the method (n = 125)

Rank Group	*Requirement*	*Medium Rank*	*Relevance**	*U2*	*U5*	*U3*	*U4*	*U1*	*U6*
1	U2	2,82	1,6		0,457	0,214	● 0,003	● 0,002	● 0,000
2	U5	3,33	1,6	0,457		1,000	1,000	1,000	0,058
	U3	3,4	2	0,214	1,000		1,000	1,000	0,139
3	U4	3,69	2,3	○ 0,003	1,000	1,000		1,000	1,000
	U1	3,74	2,1	○ 0,002	1,000	1,000	1,000		1,000
	U6	4,02	2	○ 0,000	0,058	0,139	1,000	1,000	

● Requirement (row) significantly more relevant than requirement (column)
○ Requirement (row) significantly less relevant than requirement (column)
0,1 corrected p-value from pairwise post-hoc test
*Relevance of the requirement [1(essential) - 5(very unimportant)]

Interpretation of the requirements

To develop the intended method to enable continuous improvement of distributed collaboration, the identified requirements were transferred into measures, elements and activities of the method. First focusing on the requirements of the support evaluation, the requirements were interpreted and operationalized as activities of the EDiT method with different goals that can be comprised in four consecutive phases (Figure 1).

The first phase of the method is the potential analysis, in which the situation and problem analysis take place. Therefore, potentially critical activities of the distributed product development process are identified and potential fields of action of the distributed collaboration that offer potential for improvement are analyzed. In the measure definition (phase 2), alternative solutions as measures for addressing the potentials are derived, prioritized, and selected. The third phase focuses on the measure implementation and thus, serves to develop the identified potentials based on the selected measures. In the last phase, the measure evaluation, the effort and benefit of the measures are compared to evaluate the success of the measure and to be able to give a statement on all evaluation criteria after Blessing and Chakrabarti (2009) focusing the success evaluation.

Figure 1 represents an initial reference process model of the EDiT method focusing on the individual possibilities of applying the method. The different phases with their activities can be carried out iteratively employing various possibilities of implementation. Workshops, interviews and surveys are only a small overview of implementation possibilities that can be selected individually in the different phases. This can lead to a low-barrier application of the method in practice due to the possibility to choose the method implementation suitable for the respective development context (Duehr et al., 2019). For example, a retrospective to identify improvement potentials and measures to address the potentials can support the first and second phase of the EDiT method (Duehr, Kavakli, & Albers, 2021).

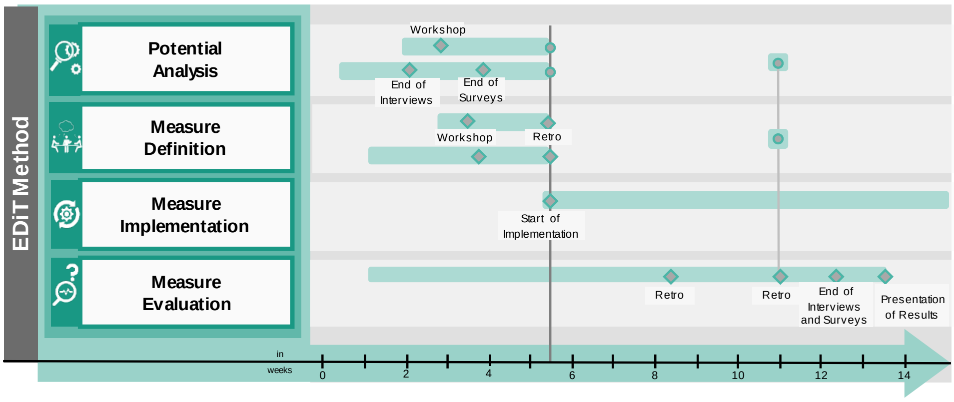

Figure 1 Reference process model of the EDiT method

In the following research activities, the remaining requirements of the application and success evaluation will be finally transferred to the individual phases as measures, elements and activities. This will include, for example, subdividing the phases of the method into steps presenting an individual package of activities. Each activity is assigning clear instructions for action and possibilities for implementation. Another aim is to ensure that the final representation of the method corresponds as closely as possible to the underlying procedure. In addition, the method should offer the possibility to respond to the individual needs of different teams and therefore be applicable in different teams without compromising existing processes by enabling individual method selection and implementation possibilities in the different phases and steps.

5 Summary and discussion

The main achieved scientific contribution is the identification of relevance weighted user requirements of a method for enabling continuous improvement of the collaboration of distributed development teams as well as the operationalization of the initial requirements in a holistic method. As the relevance rating for all requirements was at a similar and very high level it is considered that all requirements should be taken into account in the development of the method. The initial method is based on needs derived from potential users as the center of product development (Albers et al., 2019). Therefore, this contribution built the foundation to further develop a user-centered method that supports distributed development teams to continuously improve their collaboration while considering their individual development context. In addition, the procedure for developing the method and the initial representation is presented to ensure transferability to other research areas.

Limitations

Several limitations must be noted. First, the first stage of the Delphi study was only conducted with five experts. However, when selecting the experts, attention was paid to covering various criteria such as the domain of product development, function in the team or years of experience. Nevertheless, it cannot be assumed that every potential requirement has been identified. Second, since all requirements were assessed as relevant, not all requirements could be implemented in the initial method. Thus, the focus of this contribution was on implementing the requirements that were assigned to the support evaluation. And third, as a result, there is still a lack of a guideline that supports the planning and the application of the method for the individual development context.

Practical implications

By focusing on the needs of the development team from the very beginning and by presenting an initial representation of the method, a low-barrier application of the method in practice with a high probability of success is enabled. Keeping the phases of the method generic allows the development team to further adapt the method to their individual development context. Nevertheless, the guideline to be developed in the next research activities will support the implementation and application of the method. Eventually, this will positively contribute to an improved collaboration in distributed product development processes.

Directions for future research

This study built the foundation for a user-centered method that enables continuous improvement of distributed development teams. Coming from an initial operationalization of the identified requirements and the initial representation of the method, a detailed operationalization will take place. Further research will focus on the development of a guideline that supports development teams to adapt and implement the method to their individual development context. Moreover, to provide a method that is to be transferred to different development contexts, a comprehensive validation of the method will be carried out. Therefore, to deal with the trade-off between internal and external validity (Roe & Just, 2009), the validation of the method will take place with field experiments as well as laboratory such as the Engineering simulator (Hofelich et al., 2021). This enables an iterative validation and subsequent adjustment of the method to ultimately provide a promising and transferable method to distributed development teams.

References and Notes

Ahuja, J. (2017). Modelling the Success Factors of Virtual Teams. *Indian Journal of Science and Technology*, *9*(48). https://doi.org/10.17485/ijst/2016/v9i48/105874

Albers, A., Weissenberger-Eibl, M. A., Duehr, K., Zech, K., & Seus, F. (2020). Literature-based identification of success-relevant influencing factors of distributed product development. In *Procedia CIRP: Enhancing design through the 4th Industrial Revolution Thinking* (91st ed., Vol. 91). https://doi.org/10.1016/j.procir.2019.11.007

Albers, A., & Braun, A. (2011). A generalised framework to compass and to support complex product engineering processes. *International Journal of Product Development*, *15*(1/2/3), Article 43659, 6. https://doi.org/10.1504/IJPD.2011.043659

Albers, A., Heimicke, J., Spadinger, M., Degner, N., & Duehr, K. (2019). The Product Developer in the Centre of Product Development: A Systematic Literature Review on

Describing Factors. *Proceedings of the Design Society: International Conference on Engineering Design*, *1*(1), 1843–1852. https://doi.org/10.1017/dsi.2019.190

Albers, A., Klingler, S., & Ebel, B. (2013). Modeling systems of objectives in engineering design practice. In *Proceedings of the 19th International Conference on Engineering Design (ICED13): Design For Harmonies*, Seoul, Korea.

Albers, A., Reiss, N., Bursac, N., & Richter, T. (2016). iPeM – Integrated Product Engineering Model in Context of Product Generation Engineering. *Procedia CIRP*, *50*, 100–105. https://doi.org/10.1016/j.procir.2016.04.168

Badke-Schaub, P., Daalhuizen, J., & Roozenburg, N. (2011). Towards a Designer-Centred Methodology: Descriptive Considerations and Prescriptive Reflections. In H. Birkhofer (Ed.), *The future of design methodology* (pp. 181–197). Springer. https://doi.org/10.1007/978-0-85729-615-3_16

Bavendiek, A.-K., Huth, T., Inkermann, D., Paulsen, H., Vietor, T., & Kauffeld, S. (2018). Collaborative Design: Linking Methods, Communication Tools and Competencies to Processes. In *DESIGN 2018 - 15th International Design Conference*. https://doi.org/10.21278/idc.2018.0420.

Bavendiek, A.-K., Inkermann, D., & Vietor, T. (2017). Interrelations Between Processes, Methods and Tools in Collaborative Design - a Framework. In *Proceedings of the 21st International Conference on Engineering Design (ICED 17)*, Vancouver, Canada.

Birkhofer, H., Jänsch, J., & Kloberdanz, H. (2005). An extensive and detailed view of the application of design methods and methodology in industry. In *DS 35: Proceedings ICED 05, the 15th International Conference on Engineering Design*, Melbourne, Australia.

Blessing, L. T.M., & Chakrabarti, A. (2009). *DRM, a Design Research Methodology*. Springer London. https://doi.org/10.1007/978-1-84882-587-1

Bullinger, H.-J., Wörner, K., & Prieto, J. (1997). *Wissensmanagement heute: Daten, Fakten, Trends; Ergebnisse einer Unternehmensstudie des Fraunhofer-Instituts für Arbeitswirtschaft und Organisation in Zsarb. mit dem Manager Magazin*. Fraunhofer Institute for Industrial Engineering IAO. https://doi.org/10.1007/978-3-642-71995-0_2

Dillenbourg, P., Baker, M., Blaye, A., & O'Malley, C. (1996). The evolution of research on collaborative learning. In P. Reimann & H. Spada (Eds.), *Journals of related interest. Learning in Humans and Machine: Towards an interdisciplinary learning science* (1st ed., pp. 189–211). Elsevier; Pergamon.

Dörner, D. (1976). *Problemlösen als Informationsverarbeitung*. Kohlhammer; Stuttgart.

Dorst, K. (2006). Design Problems and Design Paradoxes. *Design Issues*, *Vol. 22*(3), 4–17. https://doi.org/10.1162/desi.2006.22.3.4

Duehr, K., Hirsch, M., Albers, A., & Bursac, N. (2020). A Methodology to Identify and Adress Improvement Potentials in Communications Processes of Distributed Product Development - An Initial Approach. *Proceedings of the Design Society: DESIGN Conference*, *1*, 541–550. https://doi.org/10.1017/dsd.2020.35

Duehr, K., Efremov, P., Heimicke, J., Teitz, E. M., Ort, F., Weissenberger-Eibl, M., & Albers, A. (2021). The Positive Impact of Agile Retrospectives on the Collaboration of Distributed Development Teams – a Practical Approach on the Example of Bosch Engineering GmbH. In Design Society (Chair), *Proceedings of the 23nd International Conference on Engineering Design (ICED21),* Gothenburg, Sweden. (accepted)

Duehr, K., Heimicke, J., Breitschuh, J., Spadinger, M., Kopp, D., Haertenstein, L., & Albers, A. (2019). Understanding Distributed Product Engineering: Dealing with Complexity for Situation- and Demand-Oriented Process Design. In *CIRP Design Conference.* Symposium conducted at the meeting of CIRP, Póvoa de Varzim, Portugal. https://doi.org/10.1016/j.procir.2019.04.200

Duehr, K., Kavakli, E., & Albers, A. (2021). Improving distributed collaboration – methods for identification and developing of improvement potentials. In Design Society (Chair), *Proceedings of the 23nd International Conference on Engineering Design (ICED21),* Gothenburg, Sweden. (accepted)

Dumitrescu, R., Albers, A., Riedel, O., Stark, R., & Gausemeier, J. (2021). *Advanced Systems Engineering: Wertschöpfung im Wandel*. https://www.advanced-systems-engineering.de/documents/20210414_ASE_Engineering_in_Deutschland.pdf

Ehrlenspiel, K., & Meerkamm, H. (2013). *Integrierte Produktentwicklung: Denkabläufe, Methodeneinsatz, Zusammenarbeit* (5th ed.). Hanser, München. https://doi.org/10.3139/9783446436275

Gaul, H.-D. (2001). *Verteilte Produktentwicklung - Perspektiven und Modell zur Optimierung.* Verlag Dr. Hut, München.

Gericke, K., Eckert, C., & Stacey, M. (2017). What do we need to say about a design method. In *Proceedings of the 21st International Conference on Engineering Design (ICED 17)*, Vancouver, Canada.

Gericke, K., Meißner, M., & Paetzold, K. (2013). Understanding the context of product development. In *Proceedings of the 19th International Conference on Engineering Design (ICED13): Design For Harmonies,* Seoul, Korea.

Gierhardt, H. (2002). *Global verteilte Produktentwicklungsprojekte: Ein Vorgehensmodell auf der operativen Ebene*. Zugl.: München, Techn. Univ., Diss. (1. Aufl.). *Produktentwicklung*. Verl. Dr. Hut.

Häder, M., & Häder, S. (2000). Die Delphi-Methode als Gegenstand methodischer Forschungen. In M. Häder (Ed.), *ZUMA-Publikationen. Die Delphi-Technik in den Sozialwissenschaften: Methodische Forschungen und innovative Anwendungen* (pp. 11–31). Westdt. Verl. https://doi.org/10.1007/978-3-663-09682-5_1

Herbsleb, J. D., Mockus, A., Finholt, T. A., & Grinter, R. E. (2000). Distance, dependencies, and delay in a global collaboration. In W. Kellogg (Ed.), *Proceedings of the 2000 ACM conference on Computer supported cooperative work* (pp. 319–328). ACM. https://doi.org/10.1145/358916.359003

Hofelich, M., Bursac, N., Mantel, K. V., Omidvarkarjan, D., Schneider, T., Meboldt, M., & Matthiesen, S. (2021). Engineering Simulator as Research Environment for design support – Literature review-based attributes of research environments and investigation of agile product development. In *31th CIRP Design Conference*, Twente, The Netherlands. (accepted)

Jänsch, J. (2007). *Akzeptanz und Anwendung von Konstruktionsmethoden im industriellen Einsatz: Analyse und Empfehlungen aus kognitionswissenschaftlicher Sicht*. Zugl.: Darmstadt, Techn. Univ., Diss. *Berichte aus dem Fachgebiet Produktentwicklung und Maschinenelemente der TU Darmstadt: Vol. 396*. VDI-Verl.

Kirchner, E. (2020). *Werkzeuge und Methoden der Produktentwicklung*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-61762-5

Konradt, U., & Hertel, G. (2002). *Management virtueller Teams: Von der Telearbeit zum virtuellen Unternehmen*. *Management und Karriere*. Beltz.

Krause, F.-L. (Ed.). (1998). *New tools and workflows for product development: Proceedings ; Production Technology Centre Berlin, Germany*. Fraunhofer-IRB-Verl.

Kuster, J., Bachmann, C., Huber, E., Hubmann, M., Lippmann, R., Schneider, E., Schneider, P., Witschi, U., & Wüst, R. (2011). *Handbuch Projektmanagement: Agil - Klassisch - Hybrid* (4th ed.). Springer.

Larsson, A., Törlind, P., Karlsson, L., Mabogunje, A., Leifer, L., Larsson, T., & Elfström, B.-O. (2003). Distributed Team Innovation - A Framework for Distributed Product Development. *Proceedings of ICED 03, the 14th International Conference on Engineering Design, Stockholm*.

Lindemann, U. (Ed.). (2016). *Handbuch Produktentwicklung*. Hanser. https://doi.org/10.3139/9783446445819

Lipnack, J., & Stamps, J. (1997). *Virtual teams: Reaching across space, time, and organizations with technology*. Wiley.

Marxen, L. (2014). *A Framework for Design Support Development based on the integrated Product Engineering Model iPeM* [Dissertation]. Karlsruher Institut für Technologie, Karlsruhe.

Pohl, K. (2007). *Requirements engineering: Grundlagen, Prinzipien, Techniken* (1. Aufl.). dpunkt Verlag, Heidelberg.

Roe, B. E., & Just, D. R. (2009). Internal and External Validity in Economics Research: Tradeoffs between Experiments, Field Experiments, Natural Experiments, and Field Data. *American Journal of Agricultural Economics*, *91*(5), 1266–1271. https://doi.org/10.1111/j.1467-8276.2009.01295.x

Schmidt, K. (2017). *Vertragsbasierte Koordination verteilter Produktentwicklungsprozesse*. Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-17241-1

Schömann, S. O. (2012). *Produktentwicklung in der Automobilindustrie: Managementkonzepte vor dem Hintergrund gewandelter Herausforderungen*. Zugl.: Eichstätt-Ingolstadt, Univ., Diss., 2011 (1. Aufl.). *Gabler Research*. Gabler Verlag / Springer Fachmedien Wiesbaden GmbH. https://doi.org/10.1007/978-3-8349-6673-5

Schuh, G. (2012). *Innovationsmanagement*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-25050-7

Stöger, G., & Thomas, G. (2007). *Teams ohne Grenzen: Und es geht doch: virtuelle Teams erfolgreich führen*. Orell Füssli, Zürich.

Welp, E. G. (1996). Planung und Steuerung verteilter Produktentwicklungsprozesse. *VDI-Berichte Band 1270 S. 185- 208, VDI-Verlag, Düsseldorf*.