A Perspective from Large vs Small Companies Adoption of Agile Methodologies

Manuela Petrescu[a] and Simona Motogna[b]
Department of Computer Science, Babes Bolyai University, Cluj-Napoca, Romania

Keywords: Agile, Small Companies, Start-Ups, Survey, Adapt, Challenges, Large, Adoption, Adaptation.

Abstract: To offer a perspective from large versus small companies' adoption of Agile Methodologies we performed a study in which we discussed and analyzed the answers of 31 interviewees from 14 companies. We carefully selected the data set to be representative for all the actors in the market from the company's size, business model, role balance, and gender balance. We found out that most of the companies adopted the Agile methodologies in a more or less extensive manner. By far, the most used ceremony was daily meetings, other ceremonies were sometimes merged or removed from the methodology implementation. We also investigate how companies adopted the Agile mindset by analyzing how the Agile Manifesto was implemented in these companies, even if it was not in the scope of our study. From a scientific point of view, this study contributes from a methodological perspective to a better understanding of Agile adoption in different types of companies and of the challenges related to Agile practices and processes. For practitioners, the results presented in this paper offer evaluation means and solutions for Agile adoption.

1 INTRODUCTION

As the applications become more complex, new methodologies emerged to fasten delivery. Agile appeared as a response to different requirements that could not be fulfilled through plan-driven methodologies. Agile methodology started to be widely used because it offers more flexibility and responsiveness to change; it offers support to continuous changes and delivery, even if it has its own downsides (Brown, 2013). SCRUM, XP (Extreme Programming), and FDD (Feature Driven Development) methodologies imposed themselves and are analyzed and compared in different papers (Akhtar et al., 2022; Saleh et al., 2019).

In our opinion, small companies are the engines for innovation, they are forced to bring new ideas, and new approaches to survive and to get a market share. They are more dynamic and flexible, being able to change processes on the run much faster compared to large-scale companies. The key to future development approaches is to analyze how software development start-ups are implementing Agile methodologies. Knowing how small companies adopt Agile Methodologies compared to medium and large companies provide useful information for other companies doing business in the same sector. Thus, the motivation of this paper is strongly related to the business environment and the paper could provide useful information to other companies working in the same business domain.

The transition of IT projects to agile practice is not always easy (Burga et al., 2022); large companies faced issues in adopting Agile in the development process (Klunder et al., 2019) and startups use a hybrid method - Agile and Waterfall (Mishra et al., 2021). Some research was conducted related to various topics for Agile adoption in start-ups: the use of Agile practices (Souza et al., 2019), internal communication (González-Cruz et al., 2020), process metrics usage (Choras et al., 2020). This study addresses a new perspective: how the small companies adapted the team structure, the roles, the processes in the SCRUM teams, and the challenges they faced in adopting and adapting Agile methodologies. Despite the maturity of Agile practices and the considerable number of research contributions, there are organizational aspects that need a deeper understanding: (i) adaptation of Agile practices and (ii) challenges faced by the organization during the implementation of the Agile framework. This study aims at comparatively

[a] https://orcid.org/0000-0002-9537-1466
[b] https://orcid.org/0000-0002-8208-6949

Petrescu, M. and Motogna, S.
A Perspective from Large vs Small Companies Adoption of Agile Methodologies.
DOI: 10.5220/0011716800003464
In *Proceedings of the 18th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2023)*, pages 265-272
ISBN: 978-989-758-647-7; ISSN: 2184-4895

responding to these open issues, by exploring differences between large and small companies. We use empirical methods, starting with interviews of people with management roles to find out the company's approach, then we used open coding to process the obtained data and to produce a qualitative analysis. The paper is structured as follows: Preliminaries and Related Work sections analyze the current studies, and the Data collection and analysis section describes how we obtained and analyzed the data. Results are detailed in the Findings through empirical investigation section, and we conclude in the Conclusions and future work section.

2 PRELIMINARIES

Agile addresses some of the waterfall methodology issues as it adapts to requirements changes, offering a flexible development approach focused on customer satisfaction and team involvement. The values and principles of the Agile methodologies were defined in Agile Manifesto (Kent Beck, 2001), in which the authors stated four core values to assure an open-to-change process. They stated the importance of *Individuals and interactions over processes and tools, of working software over comprehensive documentation, Customer collaboration over contract negotiation, and Responding to change over following a plan*. Several development methodologies follow these principles, the most used being: Scrum, Kanban, XP, and FDD. We decided to use the definition for small companies provided by the Commission of the European Communities (the European Communities,), which states that a small enterprise has between 10 to 49 persons employed and an annual turnover of not more than €10 million.

3 RELATED WORK

A considerable number of papers were written over the last 20 years addressing Agile methodologies, adoption to different domains, and case studies. In this study, we focused on contributions related to Agile adoption in IT companies. Agile was not adopted only in IT companies but it spread in other domains as well, modified according to their needs and scaled (Gerster et al., 2020). In IT it started in companies that develop specific niche software products (automotive (Katumba and Knauss, 2014) or networking (Smits and Rilliet, 2011)). The authors (Smits and Rilliet, 2011) spent six months in a center for Agile practice adoption at Cisco Voice Technology and they concluded that the teams should get Agile skills before *"attempting to change the Software Development Life Cycle"*. In Nokia's case (Laanti et al., 2011) the authors concluded that *"Agile is here to stay"* after gathering and analyzing more than 1000 answers from people working there in an Agile methodology; people mentioned a higher satisfaction, increased quality and transparency, as earlier detection of defects; 60% stated that prefer working based on agile methods. The same conclusion is drawn from (Klunder et al., 2019) – a literature review study based on 85 papers on agile transformation in large companies. The authors also proposed an agile transformation model based on the reviewed literature and provided an example for the automotive domain. The Agile adoption success in large companies depends on a set of factors such as company culture, support from management, change leaders, prior Agile experience (Kalenda et al., 2018), and communication style and methods (Martini and Bosch, 2013). Adopting a new methodology raises some challenges, mostly related to people's resistance to change, quality assurance concerns, and integration with existing processes (Kalenda et al., 2018). Other authors were interested in Agile adoption in start-ups and small companies, what Agile methodologies are mostly used (Souza et al., 2019), and how agile internal communication works (González-Cruz et al., 2020).

According to our knowledge, most of the research focused on Agile adoption in large companies, so we consider that a comparison between Agile methodology adoption in large versus small companies is an interesting topic and the results can be used by other actors from the same business environment.

4 DESIGN OF THE STUDY

Large companies that have a portfolio of clients find it easier to survive and to assure a cash flow. Small companies need to innovate, to be more flexible to client's needs, and to adapt faster to new methodologies. Companies can adapt to the challenges by changing the structure of the teams, the processes, the people's roles, and their attributions. In this study, we propose to see how small companies adapted Agile Methodologies and to perform a comparison of how Agile methodologies are used in small companies, respectively large companies adoption. In conclusion, we answer the following questions:

A. How is the team structure adapted in small companies? / How does the team structure differs in small vs large companies?

B. How are the processes and the people's roles adapted in small vs large companies?

C. How are the challenges perceived in small companies vs large companies?

5 DATA COLLECTION AND ANALYSIS

We opted for an empirical investigation and we used qualitative methods. We performed the interview studies according to the requirements and standard process as described in (Ralph, Paul (ed.), 2021). In our study we had conversations with one participant at a time, we recorded the interviews, then we applied quantitative data analysis to the participant's answers. Meeting these criteria classifies our study as a qualitative survey according to ACM Standards (Ralph, Paul (ed.), 2021). The major concern was that the data set had to be representative for the selected topic. Due to this aspect, we tried to have a highly diverse participant set in terms of position, company, and gender. We interviewed 31 people from 14 companies, 14 women and 17 men. Their position was also diverse: team leaders 10, product owners 3, delivery managers 6, SM 12, chief software architect 2, and testers 5. It was interesting to see that many of the persons that were in leading positions roles (team leads, release managers, POs) or in other roles (business analyst, developer roles, testing roles) combined their role with the scrum master and/or PO role. We designed the questions and grouped them in several categories, starting with personal info, project info, and then Agile related topics as mentioned in Table 1. As

Table 1: Interview Questions.

1. Name of the participant & Current Role
2. Experience in current role & overall working experience
3. Company Name & Type [multinational, local, national]
4. Application domain, Project theme
5. Size of the project & Size of the team
6. Development period (in terms of months, years)
7. What roles are in the Agile team?
8. Do you use Agile methodology? Which parts of the methodology are you using and how you adapted it?
9. Please describe which are the major challenges related to Agile methodology implementation in your team.

we wanted to investigate the software development process in large companies compared to small companies, we had to assure that the participant set is relevant, and the participants that work in small versus large companies are equally represented. There were more than 34 hours spent in interviews and more than 52 hours spent reviewing the data and creating transcripts. The data gathering process was done according to the requirements and the standard process as described in (Ralph, Paul (ed.), 2021). Data were collected mainly as open answers to get a more profound understanding of the processes, roles, overall functioning, and to allow respondents to fully describe relevant details of interest. We coded each answer with *RSx* where x was the number of interview and S came from the interviewee company size: S (small) or L (large). We decided to use thematic analysis (Braun et al., 2019; Kiger and Varpio, 2020) to interpret the text, previously used in software engineering in other studies (Motogna. et al., 2021; Petrescu et al., 2022).

6 FINDINGS THROUGH EMPIRICAL INVESTIGATION

RQ1: *How is the team structure adapted? / How does the team structure differs in small vs large companies?*

Team structure as defined in the SCRUM should consist of a Product Owner (PO), a SCRUM Master (SM), and the SCRUM team, such that the team has all the resources and the knowledge to be able to deliver each sprint. Once the decision to use the SCRUM methodology is used, people need to be allocated to fill up the roles. We asked each interviewed person about the PO role in their team and to describe how SCRUM was implemented. We found out that *"We changed the role responsibility (PO tasks were taken over by Scrum Master/Team lead"* (RL31). More interviewers mentioned that the backlog changed during the sprint without the team's acceptance. In many cases, the PO had a hard time negotiating with the client, and most of the time the team had to complete tasks with higher priority in the same sprint. The interviewees mentioned that there are issues with *"backlog prioritization (tasks appear with 0 priority and they need to be solved and impact the sprint, as no tasks are taken out). The prioritization is done by PO and BAs."* (RL17).

In our study, we found out that the structure of the SCRUM teams is very different, it depends on the size of the project, the project type, the client's requirements, and the involvement. For example, we encountered the following situations for the PO role: no explicit PO allocated (2 cases), PO assigned by the client (12 cases), PO assigned by the development team (11 cases), and PO shared (5 cases). In our interviews, only one small company had a person designated to do a PO role, but we had a complex project in which there were 7 persons working

as PO: *"We had seven POs and no SCRUM master"* (RL25). Another observation, the shared PO role appeared in large companies for small teams, and there were only two cases in a large company where the PO role was not explicitly allocated (due to project specificity). For the PO role, there are no significant

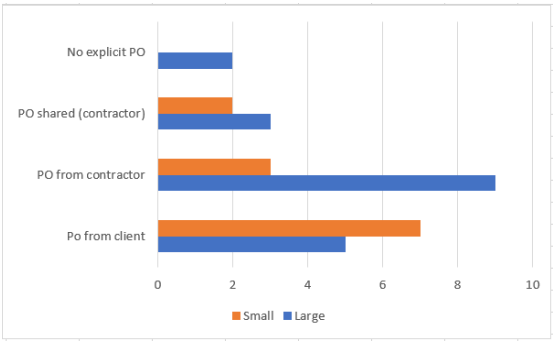

Figure 1: PO Distribution in small vs large companies.

differences between large and small companies; for SM role, the situation is different. We found out that the SM role was more adapted to each project's specificity (size, niche):
- a person was designated only for a project in a) large companies or b) small companies but large projects (15 cases)
- shared SM, the same person had this role in multiple projects (11 cases)
- no designated person (5 cases)
In the last case, the role was performed by PO, teamlead or by different members of the team depending on the sprint *"it was not a permanent role, for technical sprints, it was a technical person, for a sprint with polishing, design, the role was taken by PO"* (RS12).

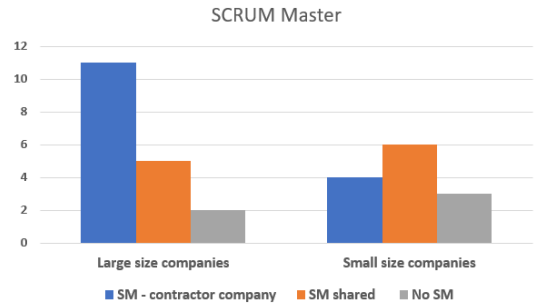

Figure 2: SCRUM Master Distribution.

Non-Coding Roles. All teams had developers for programming roles, and most of the teams had QA, but as for the other roles: business analyst, DevOps, and subject matter experts, things were more flexible and they were adapted based on the required skills, client, and project. We found out that some roles (DevOps) were filled based on demand from interval resources or by contracts when there was no internal resource: *"There was a need for technical experts (for cloud migration)- they were request based and contracted"* (RS12). Business analysts or subject matter roles were sometimes fulfilled by PO, but in some cases, independent contractors were hired for domain know-how *"It depends on the problem, for a business problem sometimes they hired a consultant"* (RL24).

We found out that small companies were more likely to find external resources for technical roles; for business roles (PO, SM), sometimes the client was involved. Other non-coding roles (i.e. technical writer) were not mentioned; practically, there was almost no effort related to writing project documentation. Large companies hired specialized personnel for documentation writing, and all companies (small and large) used internal resources when a presentation or other internal documents had to be written. Usually, large companies had internal resources for technical roles such as DevOps or site engineer, allocated project based.

Our analysis revealed the following key points:
A. There is no significant difference in non-coding roles between small or large companies, they rely on external resources (usually contract based) when they don't have the necessary know-how. Most companies try to use as few resources as possible for tasks such as documentation (technical writers' roles are mentioned briefly only in large companies).
B. In small companies, the team structure is more flexible, as PO and SM roles are performed by different team members; in large companies, there are more dedicated persons to perform these roles.

RQ2: *How was implemented Agile Manifesto? How are the processes and the people's roles adapted in small vs large companies?* We structured the answer to this question in two parts, the first one related to how the principles in Agile Manifesto were adapted, and the second related to the adaptation of processes and people's roles.
RQ2.1:*How were adapted Agile Manifesto principles?*

Individuals and Interactions over Processes and Tools. Agile enforces interactions and teamwork over processes or specific usage of tools. In the interviewee's perception, implementing all the ceremonies (or most of them) introduces high costs. *"Spending a day each fortnight it's a lot of wasted time. "* (RS24) or *"processes (spring defining, planning, retrospective are time-consuming – 15% time can be consumed only on these tasks)"* (R26). Some

teams modified the ceremonies, shorted, merged, or even removed them to save time *"retrospective and planning were in the same meeting – all in one"* (RL26). However, daily meetings were the one ceremony that was the most present and suffered less adaptation. Some companies kept the main idea and tasks, but the processes and the responsibilities were adapted, *"there was a mini squad team to review the document"* (RL11).

Working Software over Comprehensive Documentation. To have comprehensive documentation, each team should assign people to write it. In our interviews, none of the interviewees mentioned technical writers, nor did they mention writing documentation tasks, client's manuals, or other types of documentation. Everyone's focus was on delivering working software and sometimes on fixing and supporting testing versions (cases when the backlog was modified due to high-priority bugs). As all the companies valued working software more compared to comprehensive documentation, and the processes were not related to the degree of Agile methodology adoption, we believe that the approach is due more to the business perspective, as the clients don't accept the costs of such comprehensive documentation. Some clients had a hard time accepting the cost of different development tasks, *"clients do not understand the impact in terms of the cost of their CRs (change requests);"* (RL2).

Customer Collaboration over Contract Negotiation. Customer collaboration is strong, in our study, 35% of the projects had the Product Owner named by the client (11 projects from 31 interviewed). In 1.16% (5 of 31) projects, the client preferred not to interfere at all in the development phase, so those projects can be considered more contract related. The rest of the projects had flexible length sprints and the backlog was updated based on the client's input. In conclusion, we can state that customer collaboration was extensive in most of the projects (83% of them).

Responding to Change over Following a Plan. Developing a project requires a plan, and the architecture skeleton should be in place before starting to implement the other features. The main topic is in fact how easily the backlog artifact can be adapted to new requirements, to changes of prioritization, or based on client feedback. Only 4 interviewees mentioned that it was plan-driven development, one interviewee did not mention it. Other interviewees mentioned that there were issues due to work overload: topics and tasks were added to the sprint backlog during the sprint. We concluded that in most of the projects, 83% of the total number of projects, the methodology was to respond to change, *"support tickets were resolved with higher priority than the sprint tasks"* (RL22), *"Agile in fact is a waterfall split in sprints. There are deadlines, a list of features, but the work is structured in sprints."* (RS12).

RQ2.2:*How are the processes and the people's roles adapted in small vs large companies?*

The answer to this research question is obtained by analyzing the results obtained through open coding of the responses to question 13 (see Table 1). The procedure followed in the coding process considered the already existing themes/categories related to Agile methodology and project management and keywords generated from data analysis have been classified according to these themes. Two members of the study have previous industrial experience in Agile, one with over 15 years working in IT company, while one member had significant experience in Grounded Theory application to software engineering problems.

Table 2 contains the main categories (themes) obtained through coding, keywords (labels) associated with this category, and respectively suggestive examples for these keywords. The process was applied for all answers, regardless of the company type. We then explore which is the distribution of these categories, depending on the size of the company, as depicted in Figure 3.

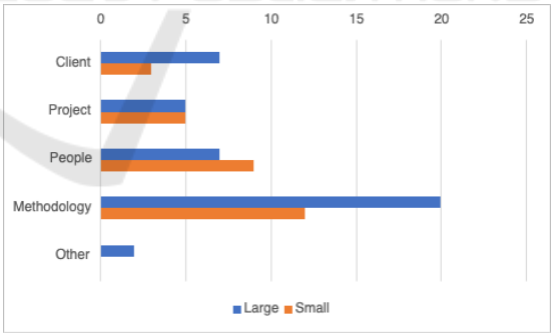

Figure 3: Adapting Agile in large vs. small companies.

Our analysis revealed the following **key points**:

A. The majority of adapting changes target the methodology itself both for small and large companies in terms of flexible ceremonies and adapted artifacts; in some cases, companies decided to drop SCRUM in favor of Kanban. The adaptations seem to have similar distribution for small and large companies.

B. In the case of small companies, adapting Agile was targeting people on a bigger scale than in the case of larger companies. Having *"mixed roles"* and consid-

Table 2: Categories for "Adapting SCRUM/Agile"

Category	Topics	Examples
Client	client dependency, changes, client only partially involved	*"frequent client requirements to modify scope", "client could ask for change requests during the sprint", "client not present to define all subtasks"*
Project	project, modify scope	*"was adapted depending on project needs", "Kanban and Scrum are combined due to development purpose"*
People	team availability, human resources, mixed roles	*"adapted to facilitate the team", "roles are mixed; SM also has other roles"*
Methodology	estimations, flexible ceremonies, adapted artifact, prioritization	*"board is not always well defined, just add items", "grooming eliminated from planning", "long meetings, involving many people in which one does not bring any input/value"*
Others	non formal, enforce new ideas	*"nothing is formalized", "scaled Agile framework: one sprint for stabilization and innovation"*

ering *"team availability"* takes higher importance in small companies. We might say that adapting human resources is easier in the case of larger companies.

C. In the case of large companies, adapting changes were generated by client requirements or specificity, which might indicate that a larger company might be more sensitive to client changes, while the smaller companies will try to preserve their resources throughout the whole lifespan.

RQ3: *How are challenges perceived in small companies vs large companies?*

The answer to this research question comes from results obtained through open coding of the responses to question 14 from Table 1, following the same approach as in the case of the previous research question. We applied the same methodology of coding on all answers, regardless of the company size, which resulted in four categories of challenges, as presented in Table 3. The answers were very diverse, with some respondents nominating several challenges, and some of them even describing actions to resolve them. The Agile methodology poses a large range of challenges for both small and large companies. The majority is due to the methodology itself ($\approx 51\%$), where estimations, priorities, and planning are among the most encountered: *"people find it hard to do all the ceremonies mentioned in the methodology"* (RS16), *"priorities that had to be changed one week to another"* (RL20), *"the estimation part as is impacted by: urgent issues, new issues that must be done, or issues that were included in the sprint but can not be done due to external factors"* (RL27).

Human resources challenges were mentioned by ($\approx 31\%$) of interviewees: people are sometimes not fully allocated to a project, are not available, or is hard to find people with required skills: *"People challenges: find people with required skills"* (RL3). There are cases when there is no dedicated person for a specific role, i.e. the SM role was *"adapted to facilitate the team"*. The attributes were delegated to a specific different team member *"PO did the scrum master part", "roles are mixed"* (RL7). Small and large companies encountered other restrictions or specific conditions they had to overcome: *"Nothing is formalized"* (RL2), *"One limitation is timezone"*; to adapt: *"scaled Agile framework: one sprint for stabilization and innovation"*. They are confronted with resistance to change *"The resistance to change is big, people find it hard to do all the ceremonies mentioned in the methodology"* (RS16).

Clients or contracts also played a major role. During the phases of a project, they intervened and modified requirements or imposed a priority change in the tasks of a sprint: *"frequent client requirements to modify scope"* (RL3), *"support tickets interfered with the sprint"* (RS23). In other projects, the clients did not offer support or did not perform their tasks *"Client not present to define all subtasks"* (RL21).

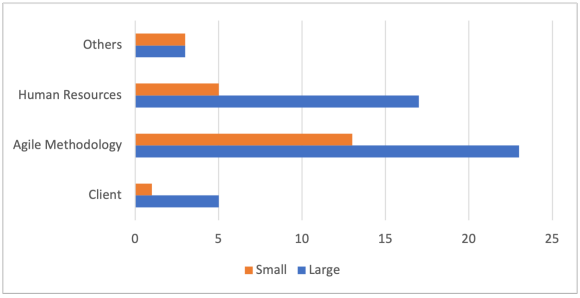

Figure 4: Challenges in large vs. small companies.

The **key points** regarding the challenges can be summarized:

Table 3: Categories for "Challenges in SCRUM/Agile".

Category	Topics	Examples
Client	client requirements, client restrictions, client interaction	*"clients don't understand the impact in terms of cost of CRs", "convince them about agile principles", "clients not answering"*
Agile	estimation, agile methodology, planning, agile implementation costs, priorities	*"when a task implementation exceeds two or three times the estimated time", "Too long meetings with too many people", "most problematic is the estimation part"*
Human resources	workload, HR shortage, communication, context switching	*"peoples with required skills", "Online communication: information passing might be affected", "Context switching impacts productivity"*
Others	project specificity, software quality, business value	*"Scrum may not be suited for a live project (in production)", "accumulated technical debt can be an issue", "lack of testing for frequent deliveries"*

A. Challenges related to the methodology itself pose the biggest problems both for large and small companies. In the case of small companies, these challenges seem overwhelming compared to the others.
B. Especially large companies struggle with human resources needs. The solution was (according to the answers) to train people from other domains to perform roles such as SM, PM, and business analyst.

7 THREATS TO VALIDITY

Being an empirical investigation, threats to validity have to be addressed. We followed existing guidelines (Ralph, Paul (ed.), 2021), and considered the following potential threats to validity: the sample size is too small, the sample size is too similar, the lack of representativeness, and the relevance of the survey based on the assumptions that it captures general people's perception. We addressed the first threat by having 31 interviews, a number close to similar studies (Burga et al., 2022; Gerster et al., 2020; Martini and Bosch, 2013; Souza et al., 2019). The sample size was diverse in terms of gender (the number of women was almost equal with the number of men), years of experience in the field, and the positions held in the company. Special attention was given to the company size as small companies and medium/large companies needed to be represented. We mitigated the last risk by asking questions strictly related to processes, where the people's perception is less important and it does not influence the answer. To ensure internal validity, we also took into consideration and minimize the impact of our interventions: we followed the same interview structure and timing and the data set was generic, so no extraneous factors can explain the results. We paid careful attention to the content of responses, trying to reduce bias by third author verification of the transcript and also explicit approval of the respondents on the final transcript. We respected the methodology of open coding, by addressing aspects related to the objectivity of the coding process and the saturation of categories. We paid attention to construct validity, we had several iterations of the questions included in the interviews. We had two interviews with two professionals with extensive experience to calibrate the questionnaire before starting the study. Finally, there was no companies or people selection bias selection. We make sure that there exists a complete anonymization of persons and companies.

8 CONCLUSIONS AND FUTURE WORK

We discussed and analyzed the answers of 31 interviewees from 14 companies to find out the challenges encountered in adopting different methodologies and how did they adapt these methodologies. To have a valid overview of the business environment, we had to select carefully the data set to be representative for all the actors in the market. The selected data set was diverse in terms of the company's size (small and medium/large companies), business model (product companies and outsourcing companies), and also in terms of the interviewed persons. Both genders were represented, people were having multiple roles (team lead, scrum master, release manager, and so on). We found out that most of the companies adopted the Agile methodologies in a more or less extensive manner. By far, the most used ceremony was *Daily meetings*, other ceremonies were sometimes merged or removed from the methodology implementation. There were only 2 of 32 projects that used "SCRUM by the book" methodology. We also analyzed how the Agile Manifesto was implemented in these companies, even if this was not the original scope of our study, but it provides relevant information related to Agile

mindset adoption.

From a scientific point of view, this study contributes to the body of knowledge related to Agile adoption in different types of companies and to better understand from methodological perspectives the challenges related to Agile practices. For practitioners, the results presented in this paper offer evaluation means and solutions for Agile adoption. For companies that do not completely adopt Agile, some action plans can be designed.

In the future, it would be interesting to find out how the Agile adaptation evolves in the mentioned companies, and how much the adaptation is correlated with project type (size, application domain, and so on), company size, or management style.

FUNDING

The publication of this article was supported by the 2022 Development Fund of the Babeş-Bolyai University.

REFERENCES

Akhtar, A., Bakhtawar, B., and Akhtar, S. (2022). Extreme programming vs scrum: A comparison of agile models. *International Journal of Technology, Innovation and Management (IJTIM)*, 2(2).

Braun, V., Clarke, V., Hayfield, N., and Terry, G. (2019). *Thematic Analysis*, pages 843–860. Springer Singapore.

Brown, D. D. (2013). *Agile User Experience Design*. Morgan Kaufmann.

Burga, R., Spraakman, C., Balestreri, C., and Rezania, D. (2022). Examining the transition to agile practices with information technology projects: Agile teams and their experience of accountability. *International Journal of Project Management*, 40(1):76–87.

Choras, M., Springer, T., Kozik, R., Lopez, L., Martinez-Fernandez, S., Ram, P., Rodriguez, P., and Franch, X. (2020). Measuring and improving agile processes in a small-size software development company. *IEEE access*, 8:78452–78466.

Gerster, D., Dremel, C., Brenner, W., and Kelker, P. (2020). How enterprises adopt agile forms of organizational design: a multiple-case study. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, 51(1):84–103.

González-Cruz, T. F., Botella-Carrubi, D., and Martínez-Fuentes, C. (2020). The effect of firm complexity and founding team size on agile internal communication in startups. *International Entrepreneurship and Management Journal*, 16(3):1101–1121.

Kalenda, M., Hyna, P., and Rossi, B. (2018). Scaling agile in large organizations: Practices, challenges, and success factors. *Journal of Software: Evolution and Process*, 30(10):e1954.

Katumba, B. and Knauss, E. (2014). Agile development in automotive software development: Challenges and opportunities.

Kent Beck, a. (2001). Agile manifesto. [accessed 31-July-2022].

Kiger, M. E. and Varpio, L. (2020). Thematic analysis of qualitative data: Amee guide no. 131. *Medical Teacher*, 42(8):846–854.

Klunder, J. A.-C., Hohl, P., Prenner, N., and Schneider, K. (2019). Transformation towards agile software product line engineering in large companies: A literature review. *Journal of Software: Evolution and Process*, 31(5):e2168.

Laanti, M., Salo, O., and Abrahamsson, P. (2011). Agile methods rapidly replacing traditional methods at nokia: A survey of opinions on agile transformation. *Information and Software Technology*, 53(3):276–290.

Martini, A. and Bosch, J. (2013). Communication factors for speed and reuse in large-scale agile software development.

Mishra, A., Abdalhamid, S., Mishra, D., and Ostrovska, S. (2021). Organizational issues in embracing agile methods: an empirical assessment. *International Journal of System Assurance Engineering and Management*, 12(6):1420–1433.

Motogna., S., Suciu., D., and Molnar., A. (2021). Investigating student insight in software engineering team projects. In *Proceedings of the 16th International Conference on Evaluation of Novel Approaches to Software Engineering - ENASE,*, pages 362–371.

Petrescu, M. A., Borza, D. L., and Suciu, D. M. (2022). Findings from teaching entrepreneurship to undergraduate multidisciplinary students: Case study. EASEAI, page 25–32.

Ralph, Paul (ed.) (2021). ACM Sigsoft Empirical Standards for Software Engineering Research, version 0.2.0.

Saleh, S., Huq, S., and Rahman, M. (2019). Comparative study within scrum, kanban, xp focused on their practices.

Smits, H. and Rilliet, K. (2011). Agile experience report: Transition and complexity at cisco voice technology group. In *2011 Agile Conference*, pages 274–278.

Souza, R., Rocha, L., Silva, F., and Machado, I. (2019). Investigating agile practices in software startups. In *Proceedings of the XXXIII Brazilian Symposium on Software Engineering*, pages 317–321.

the European Communities, T. C. Commission recommendation. [accessed 2022].