Adopting Collaborative Games into Agile Requirements Engineering

Adam Przybyłek and Mateusz Zakrzewski
Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology,
Narutowicza 11/12, 80-233 Gdansk, Poland
adam.przybylek@gmail.com, matzak91@gmail.com

Keywords: Collaborative Games, Innovative Games, Serious Games, Scrum, Creativity, Requirements Engineering.

Abstract: In agile software development, where great emphasis is put on effective informal communication involving diverse stakeholders, success depends on human and social factors. Not surprisingly, the Agile Manifesto advocates principles and values such as "individuals and interactions over processes and tools", "focus on the customer", "collaborate regularly", "communicate face-to-face within the team" and "have regular team introspection". However, agile methodologies have hardly provided any tools or techniques that aid the human side of software development. Additionally, more and more research suggests that customers no longer should be viewed as a passive source of information but need to be engaged in envisioning future business practice, discovering opportunities, and shaping solutions. To deal with these challenges, we propose a framework for extending Scrum with 9 collaborative games. Collaborative games refer to several structured techniques inspired by game play and designed to facilitate collaboration, foster customer involvement, and stimulate creative thinking. The feedback received from a Scrum team that leveraged our framework in two commercial projects, indicates that the adopted collaborative games: (1) make customers more willing to attend the meeting; (2) foster stakeholders' commitment; and (3) produce better results than the standard approach.

1 INTRODUCTION

Traditionally, Requirements Engineering (RE) is the process of identifying right stakeholders and eliciting their needs, documenting these needs as explicit requirements, and then, communicating and validating the requirements (Nuseibeh and Easterbrook, 2000). Stakeholders include: (1) sponsors who pay for the system, (2) end users who interact with the system to get their work done, and (3) developers who design, implement and maintain the system (Nuseibeh and Easterbrook, 2000). Hereafter, we refer to the first and second group as customers.

Przybylek (2014) enumerates a number of inherent difficulties in the requirements engineering process. Such difficulties, despite being well known, are still encountered in present industrial practice (Jarzębowicz and Marciniak, 2017). Customers rarely know what they really need (Faulk, 1997) and usually they have only a vague picture of their needs at the beginning of the project (Maciaszek, 2005; Cao and Ramesh, 2008). Moreover, their needs may be difficult to articulate (Davis et al., 2006). Furthermore, stakeholders may be numerous and distributed. Their needs may vary and conflict, depending on their perspectives of the environment in which they work and the tasks they wish to accomplish (Nuseibeh and Easterbrook, 2000). In addition, effective communication among stakeholders may be difficult as a consequence of their different vocabularies and professional backgrounds (Taylor-Cummings, 1998; Bormane et al., 2016). Moreover, the ways requirements are documented and communicated may be chosen inappropriately with respect to stakeholders' profiles (Jarzębowicz and Połocka, 2017). Finally, requirements evolve during the project partly due to exploration in the problem space, partly due to the dynamics of a business environment formed and reformed by the interactions of the stakeholders (Hoffmann et al., 2005; Redlarski and Weichbroth, 2016). As a response to some of these problems, agile methodologies were proposed and over the years have become dominant in the software industry.

In Agile software development, requirements engineering activities span the entire life cycle of a system. Thereby, the role of the customer is

Przybyłek, A. and Zakrzewski, M.
Adopting Collaborative Games into Agile Requirements Engineering.
In *Proceedings of the 13th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2018)*, pages 54-64
ISBN: 978-989-758-300-1

expanded within the entire development process by involving them in writing user stories, discussing product features, prioritizing the product backlog, and providing feedback to the development team on a regular basis (Nerur et al., 2005; Hoda et al., 2011; Bjarnason et al., 2016). This requires that the customers work with developers as active team members. The idea of having a customer as a member of a development team has grown from a single on-site customer, which has been dismissed by Kent Beck himself as "an error of early XP thinking" (Conboy et al., 2009), to a customer team "equal to or larger in size than the programming team" (McBreen, 2003). Since there is a wide range of potential customers, it would be difficult for a single person to represent them all (Ambler, 2008). In addition, in agile software development customers are expected to be collaborative and involved (Boehm and Turner, 2004). Unfortunately, agile methodologies do not provide techniques to promote these attitudes. Therefore, inadequate customer participation, inability to obtain consensus among various customer stakeholders and lack of effective knowledge sharing are still challenges confronting agile RE (Nerur et al., 2005; Cao and Ramesh, 2008; Chan and Thong, 2009; Conboy et al., 2010; Ramesh et al., 2010; Hoda et al., 2011).

In the meantime, many researchers and practitioners have acknowledged and agreed on the importance and the role of creative techniques in RE (Hoffmann et al., 2005; Maiden et al., 2010; Garnik et al., 2014; Ossowska et al., 2016). As a result, a substantial body of knowledge has been established, which can be summarized as follows. Requirements are no longer considered to exist in an implicit manner in the mind of customer stakeholders (Lemos et al., 2012), while the customers are no longer viewed as a passive source of requirements information but rather as active participants in requirements engineering process (Nguyen and Cybulski, 2008). Active participation means forward thinking, creating new visions, suggesting IT innovations, and shaping solutions (Robertson, 2005). Thus, finding the "right" requirements is not only about capturing requirements, but is instead about helping customers to discover requirements they were not aware of, and solving problems they did not know they had (Horkoff and Maiden, 2013). According to Robertson (2005), requirements analysts should invent requirements based on their understanding of the organisation's competitive business goals and context. Such requirements are not often things that requirements analysts directly asked for (Maiden et al., 2004b). Furthermore, Mahaux et al. (2013) and Svensson and Taghavianfar (2015) suggest that RE is not simply a creative process, but a collaboratively creative process, where interdisciplinary group of stakeholders work together to create ideas, solve conflicts, and reach a consensus on a novel and valuable system they want to build. Thus, traditional requirements elicitation techniques such as interviews, questionnaires, focus groups, participant observation, or document analysis are insufficient to elicit the whole range of requirements (Davis et al., 2006).

Unfortunately, agile methodologies do not provide new requirements elicitation techniques nor they explicitly support creativity. Even though Highsmith and Cockburn (2001) mention that "creativity, not voluminous written rules, is the only way to manage complex software development problems and diverse situations", agile methodologies make little reference to established creativity theories and techniques (Hollis and Maiden, 2013).

Responding to the above-mentioned challenges, we propose to equip Scrum teams with a set of serious, collaborative games. A serious game is a game whose primary purpose is not entertainment, but to solve a practical problem. A game is collaborative if two or more players must work together to achieve its goals (Gelperin, 2011). Collaborative games are designed to leverage multiple dimensions of communication that let participants engage the full power of their brains, resulting in richer, deeper, and more meaningful exchanges of information (Hohmann, 2006). At the same time, they emphasize the concepts of teamwork and collaboration which are highly valued by agile practices (IIBA, 2013). They can also bring numerous benefits to the requirement elicitation process since they typically provide immediate feedback, activate participants and increase participant's motivation (Fernandes et al., 2012; Ribeiro et al., 2014).

In our study, we selected 8 games originally introduced by Hohmann (2006; 2016) as a market research technique. Then, we adapted these games to requirements engineering activities and deployed in two commercial Scrum projects. Based on the feedback received from stakeholders who played the games according to our instructions, we proposed a framework that specifies how to integrate a set of collaborative games into the Scrum process. From a variety of agile methodologies, we chose Scrum, since it is one of the most widely adopted in industry (Rodriguez et al., 2012; VersionOne, 2017).

2 RESEARCH METHOD

The study was conducted as Action Research. In Action Research, the researcher works in close collaboration with a group of practitioners, acting as a facilitator, to solve a real-world problem while simultaneously studying the experience of solving the problem (Dawson, 2002; Davison et al., 2004). The researcher brings his knowledge of action research while the participants bring their practical knowledge and context (Baskerville and Myers, 2004). The goal of Action Research is to improve practical matters as well as to improve scientific knowledge (Baskerville and Myers, 2004). A precondition for action research is to have a problem owner willing to collaborate to both identify a problem, and engage in an effort to solve it (Easterbrook et al., 2007). The problem owner in this research was an internal software development department of the world's recognized leading food processing company with 150 years of tradition (the company wishes to remain anonymous). The department was experiencing typical challenges faced by Agile teams, such as the inability to gain access to the customer and the lack of customer involvement. Its authorities were open to new ideas and willing to deploy our framework in practice.

3 ADAPTED GAMES

3.1 Cover Story

In Cover Story (Gray et al., 2010; Hohmann, 2016; gamestorming.com), customers imagine an ideal future system so spectacular that it gets published on the front page of a newspaper. The customers must pretend as though this future has already taken place. The game encourages people to ignore all limits and "think big". As a result, it uncovers shared goals and can lead to realizing true possibilities that were once unimaginable. To play the game, the customers are divided into teams of four to six and each team is given a template (Fig. 1) that include six components:

- Cover – states the spectacular success of the software system;
- Headlines – reveal what the cover story is about;
- Sidebars – reveal interesting facets of the cover story;
- Quotes – testimonials about the accomplishment;
- Brainstorm – is used for documenting initial ideas;
- Images – pictures that support the cover story.

After taking 5 minutes for individuals to silently think over the system, the team should collaborate to fill in each component. Next, each team presents their chart.

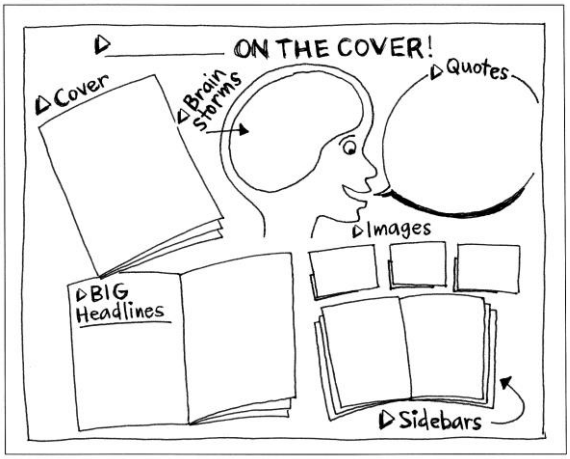

Figure 1: Cover Story (Hohmann, 2016).

3.2 Whole Product

Originally, the game aims to help the team discover new ideas about what can be done to make the product distinct and find ways to gain more customers (Levitt, 1980). However, it can be also useful for prioritizing a product backlog or for defining a product roadmap. The game board comprises four concentric circles that represent different aspects of the product (Hohmann, 2016):

- Inner Circle: Generic Product – the fundamental features that define the product;
- Circle 2: Expected Product – the features that customer considers absolutely essential;
- Circle 3: Augmented Product – the features that go beyond customer expectations;
- Outer Circle: Potential Product – everything that might be done to attract and hold customers.

Participants write ideas on sticky notes related to each circle, and then post the ideas on the chart. After all of the ideas are posted, the significance of the resulting chart is discussed. This allows developers to understand what the customers truly want from the product.

3.3 Avax Storming

AVAX Storming (Trujillo et al., 2014) is based on brainstorming. Its aim is to identify the desired functional requirements for the system. The participants write down each functional requirement on a single sticky note and place it on a flipboard. This practice helps customers to figure out the size of their project because soon the flipboard starts to be filled up. There are two note colors. One for "needed" requirements and the other for "desired" requirements. When all notes are posted on the flipboard, each requirement is explained in detail by the author and discussed by the team. Overlapped requirements are merged. Later, the notes are grouped in order to sketch the system modules. The final result is a mind map demonstrating the size of the project.

3.4 Buy a Feature

Buy-a-Feature (Hohmann, 2006) is a way of choosing the right set of features to be developed in the next Sprint. In this game, customers collaborate to purchase their most desired features. Strictly speaking, they jointly prioritize their desires as a group. Each feature should include a meaningful label, a short description, and an enumeration of benefits. Features are also assigned a price depending on their development costs and a number according to their position in the product backlog. Customers buy features that they want in the subsequent Sprint using game money. Some features may be priced so high that no single player can buy them individually. This motivates negotiations among players because they have to pool their money to buy the feature (Hohmann, 2006). Listening to the negotiations improves the understanding of what the customers really need. The total amount of money for all players involved in the exercise should allow them to purchase as many features as the developers are able to implement within a sprint.

3.5 Agile Game Incubator

This game (Hohmann, 2016; tastycupcakes.org) allows participants to teach each other the tangle of factors involved in certain dilemmas while gaining a deeper understanding of the predicament themselves. Its goal is to create a way to explain complex problems so that others will genuinely understand it and be able to form solutions. The game board consists of 5 sections (Fig. 2), representing the 5 steps of the game-creation strategy, which conveniently form the acronym PLAID (pronounced "played"). There are also colorful sticky notes that symbolize the ideas for each section:

- Problem – what you want to solve (red notes);
- Lead Objectives – what you hope to gain from solving the problem (green notes);
- Aspects – the different parts of the problem (purple notes);
- Invent – the game created to solve the problem (blue notes);
- Debrief – how the game worked out (yellow notes).

The team should brainstorms ideas related to each of the 5 steps, write them on sticky notes, and then post on the board in the respective sections.

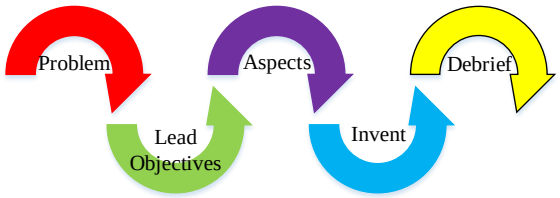

Figure 2: Agile Game Incubator.

3.6 How-Now-Wow Matrix

When people want to develop new ideas, they most often think out of the box in the creative idea generation phase. However, when it comes to convergence, people often end up picking ideas that are most familiar to them (tastycupcakes.org). The How-Now-Wow Matrix game (www.innovationgames.com) helps stakeholders select features that make the product unique and distinguish it from the competition. It naturally follows the brainstorming session, where the features that were initially flushed out are now discussed. The features are listed down on a large poster. The game board is a 2×2 matrix with "originality" on the x-axis and "feasibility" on the y-axis as shown in Fig.3. Each player is given 9 colored dot stickers (3 yellow, 3 blue, 3 green) that correspond to the quadrants of the matrix. Then, players place the respective stickers next to the three ideas that they believe are best for each category. After all the dots have been used, the number of dots under each idea are counted. The highest number of dots of a certain color categorizes the idea under that color.

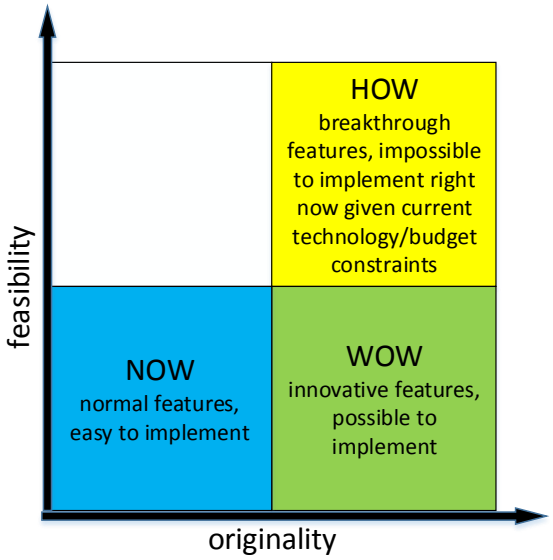

Figure 3: How-Now-Wow Matrix.

3.7 Speed Boat

Speed Boat explicitly asks customers to say what they do not like about the product. Nonetheless, it lets the facilitator stay in control of how the complaints are stated. The game starts by drawing a boat. The boat represents the software system. Everyone wants the boat to move fast. Unfortunately, the boat has a few anchors holding it back. Customers write what they don't like on an index card and place it under the boat as an anchor. The lower an anchor is placed, the more significant the issue is. Although most customers have complaints, some of them do not feel comfortable expressing their frustrations verbally, while others complain a lot about the little details. Speed Boat creates a relatively safe environment where customers can say what is wrong. By asking people to verbalize their issues in writing, the game motivates them to reflect on what is genuinely most troublesome. In this way, many of them will self-identify trivial issues as just that – trivial issues. When customers are finished posting their anchors, the facilitator reviews each one, carefully confirming the understanding of what they want to see changed in the system.

3.8 Prune the Product Tree

Prune the Product Tree helps to develop a balanced product roadmap by looking at the set of features that compose the product in a holistic manner. In this game, customers collaborate to shape the evolution of the product (i.e., the system to be developed). The product is represented by a large tree on a whiteboard (Fig. 4). Branches correspond to major areas of functionality within the software system, while leaves correspond to features. The differently colored canopies stand for various product releases. The oldest features should therefore go near the trunk. Players write a short description for each new feature on an index card, ideally shaped as leaves, and places the card on the tree. This short description generally represents a valued functionality that satisfies customers' needs. Features to add in the next Sprint are attached in the area near the edge considered as the current release. Leaves at the outer edge of the canopy are considered longer term. Participants may group leaves or draw lines between leaves to clarify relationships among features. They may also "prune" features that are not working for them by taking them off the tree.

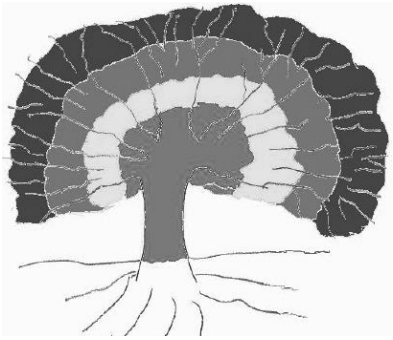

Figure 4: Prune the Product Tree (Hohmann, 2006).

4 PROPOSED FRAMEWORK

Figure 5 shows the typical Scrum life cycle with collaborative games superimposed. There are four extension points where collaborative games may occur: Product Planning, Sprint Planning, Backlog Grooming, and Sprint Review.

The purpose of Product Planning is to establish the vision of what customers wish to build and accordingly the initial Product Backlog. Three games that can support this phase are Cover Story, AVAX Storming, and Whole Product. Cover Story enables Scrum teams to understand (1) the customer's vision of the system to be developed, (2) the customers' imagination of success, and (3) how the system will create business value. In turn, Whole Product discovers features at a very high level and categorizes them into four main categories. Features belonging to the two first

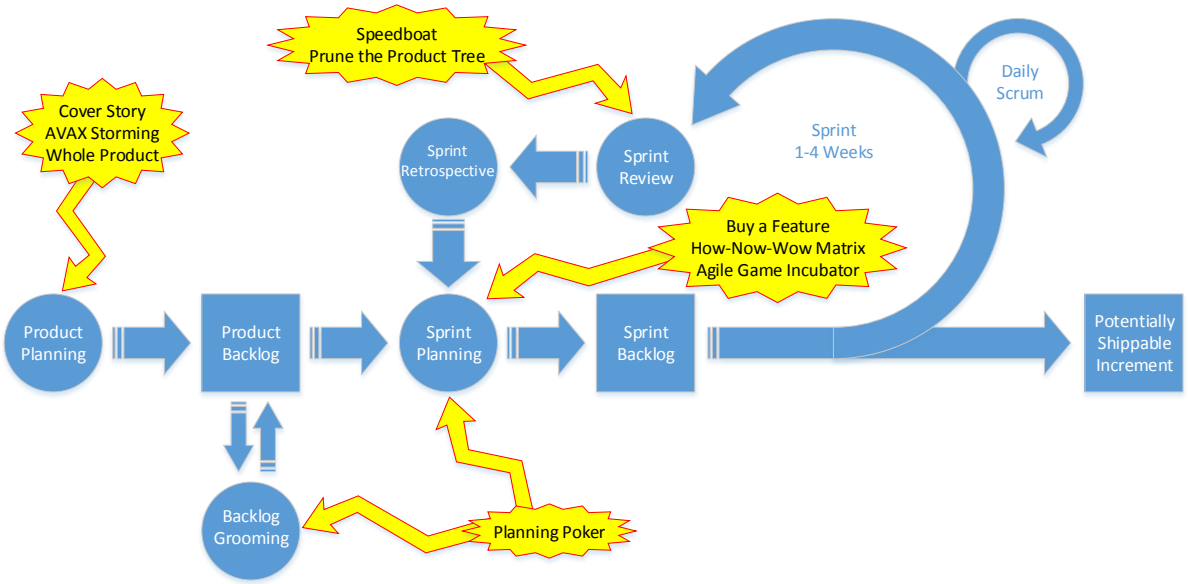

Figure 5: Scrum life cycle with collaborative games superimposed.

AVAX Storming identifies functional requirements and categorizes each as either needed or desired.

Before the start of each Sprint two consecutive meetings are held. In the first, stakeholders meet to refine and re-prioritize the Product, and to choose goals for the next iteration, usually driven by highest business value (Larman, 2003). In the second meeting, the team and Product Owner meet to consider how to achieve the goals, and to create a Sprint Backlog. The team asks enough questions that they can break down user stories of the product backlog into the more detailed tasks of the sprint backlog.

Many teams also schedule a Backlog Grooming session to prepare the Product Backlog for the Sprint Planning meeting. The intent of Backlog Grooming is to ensure that the backlog contains items that are relevant, detailed and estimated to a degree appropriate with their priority. Thereby, Backlog Grooming and Sprint Planning share the same games.

The essential game to prioritize the Product Backlog enough for the next Sprint is Buy-a-Feature. It identifies the customer's highest - priority features that can be completed within the Sprint period. The game also helps several customer representatives reach a consensus if they have conflicting interests. Likewise, How-Now-Wow Matrix aims at selecting the most valuable features as a group. On the other hand, Agile Game Incubator let the Scrum Team understand complex and unclear requirements.

Since on an agile team it is unknown who will implement the story in advance, estimating stories should be a collaborative activity for the team (Cohn, 2005). The most well-known collaborative game to provide more accurate estimates is Planning poker. Teams need to play this game at two different levels. First, there is usually an effort to initially estimate high-level user stories. Second, teams need to estimate low-level tasks that must be performed to deliver required functionality by the end of the Sprint.

The Sprint Review is held to inspect the Increment and to adapt the Product Backlog if needed. Typically, after the team demonstrates new features to the Product Owner or to the business stakeholders, all attendees collaborate on what could be done to deliver more business value to the customer. Two games – Speedboat and Prune the Product Tree – may be deploy to elicit the feedback and foster collaboration. Both games give the team the opportunity to identify those features that are simply not meeting customer needs. Speedboat solely focuses on features that need to be addressed, while Prune the Product Tree additionally provides customers with a way to indicate the directions in which to evolve the system. By observing how customers shape the tree's growth, the team has the opportunity to refine the requirements to ensure they maintain cohesion with the business.

At the end of each sprint, the team conducts a retrospective to look back at events that already have taken place, discuss what went right and wrong and decide how to improve these items for the next

sprint (Przybyłek and Kotecka, 2017; Werewka and Spiechowicz, 2017). Since the Sprint Retrospective has nothing in common with requirements engineering, in the project reported here we did not adopt games that may facilitate this meeting. However, we had done this in our previous work (Przybyłek and Kotecka, 2017), thus we refer the interested reader to that paper.

5 EVALUATION AND RESULTS

The evaluation took place in 2014 and 2015. Each game was deployed in two Scrum projects. The projects were about developing Workflow Management System. Typically, 8 stakeholders attended each game session. Among them there were 3 customers, product owner, scrum master and 3 developers. Both projects were developed by the same team but for different customers. The customers were other departments within the company.

After each game session, we issued a questionnaire. The participants were asked to indicate their level of agreement with statements about game-playing activity. The responses were on a Likert scale of: 1 – Strongly disagree; 2 – Disagree; 3 – Neutral; 4 – Agree; 5 – Strongly Agree. Table I presents average values for each game and statement across both projects and all participants. The corresponding standard deviation was always less than 1. Note, that Planning Poker was evaluated in our previous research (Przybyłek and Olszewski, 2016).

At the end of the survey, the participants were also invited to specify any additional remarks. Several of them reported a high level of enjoyment when using the games, while those who represented the customer side reported that the games were useful and motivated them to contribute to requirements elicitation.

6 DISCUSSION

Generally all games were evaluated positively, because they achieved the average score between 3.5 and 4.2. The only issue that was not appreciated was the impact on creativity, since four games obtained score below the baseline (neutral). This can be explained by the fact that both projects were designed for internal customers, so the business needs were well known and the requirements elicitation process did not require much creative thinking. In addition, the implemented software was a standard Workflow Management System and was not expected to provide any innovative features. On the contrary, willingness to attend the meeting was significantly stimulated by every game.

Whole Product, Cover Story and Agile Game Incubator performed the worst, but still above the baseline. Again, the internal customer factor probably prevented Cover Story to demonstrate its full power. In turn, Agile Game Incubator was the most difficult to understand. Indeed, it required the participants to create their own game to communicate a complex problem. How-Now-Wow

Table 1: Summary of questionnaire responses.

	Cover Story	Agile Game Incubator	AVAX Storming	Buy a Feature	How-Now-Wow Matrix	Speedboat	Prune the Product Tree	Whole Product	All games together
The game produces better results than the standard approach	3,5	3,9	4,2	4,0	3,5	4,3	3,9	3,6	3,9
The game makes customers more willing to attend the meeting	4,3	4,0	4,0	4,0	4,0	4,2	4,0	4,0	4,1
The game fosters participants' creativity	2,7	2,9	3,4	3,4	3,0	3,5	2,9	2,4	3,0
The game fosters participants' commitment	3,7	3,7	4,2	4,3	4,0	4,4	3,9	3,7	4,0
The game is easy to understand	4,0	3,6	4,4	4,4	3,9	4,5	4,1	4,0	4,1
All facets together	3,6	3,6	4,0	4,0	3,7	4,2	3,8	3,5	

Matrix also performed below expectations, probably due to a lack of innovative features in the system.

Prune the Product was considered a bit childish and its output was not perceived as meaningful even though it obtained quite high scores in all facets except creativity. On the other hand, 3 top rated games were Speedboat, AVAX Storming, and Buy-a-Feature. Each of them generated very tangible output that was considered valuable by the participants.

Note, that some games are substitutes for others, e.g. Speedboat is a substitute for Prune the Product Tree. The participants preferred AVAX Storming over Whole Product, Buy-a-Feature over How-Now-Wow Matrix, and Speedboat over Prune the Product Tree.

7 RELATED WORK

Although collaborative games are not new (Abt, 1970), to the best of our knowledge, only three studies (Gelperin, 2011; Trujillo et al., 2014; Ghanbari et al., 2015) have used collaborative games in the early stages of software development.

Gelperin (2011) defined six collaborative games that support requirements understanding by improving communication and cooperation between customers and developers. He also defined a mapping system to help developers choose the best game to play in any situation. His games could be used complementary to the games used in our framework during Sprint Planning.

Trujillo et al. (2014) proposed a game-based workshop (ActiveAction) used as an alternative for the software project's Inception phase. ActiveAction combines classical and game-based techniques to foster stakeholders' involvement and a collaborative identification of objectives, constraints and risks. Our framework shares four games with ActiveAction.

Ghanbari et al. (2015) proposed a new approach for gathering requirements from distributed software stakeholders. Their approach employs two collaborative games (Prune the Product Tree and Buy-a-Feature) provided by a web-based tool designed by Hohmann (2016).

Besides, considerable research has been directed at adopting collaborative games to support agile developers. Derby and Larsen (2006), Gonçalves and Linders (2014), Caroli and Caetano (2016), and Krivitsky (2015) presented collaborative games that can be used to facilitate retrospectives. Przybyłek and Kotecka (2017) implemented some of these games in Intel Technology Poland to make retrospectives more insightful and to avoid monotony. In turn, Przybyłek and Olszewski (2016) proposed an extension to Open Kanban, which contains 12 collaborative games that help inexperienced teams better understand the principles of Kanban.

On the other hand, numerous creativity fostering techniques have been proposed to improve the quality of requirements deliverables and to increase customer satisfaction with the final product. The most popular ones are probably brainstorming and Joint Application Development (Carmel et al., 1993). More recently, Maiden et al. (2004b) proposed RESCUE, a scenario-driven requirements engineering process that includes workshops that integrate creativity techniques with different types of use case and system context modeling. The process was successful applied to encourage creative thinking about requirements for an air traffic control system (Maiden et al., 2004a).

Mich et al. (2005) developed and evaluated EPMcreate – a creativity enhancement technique that is based on the Elementary Pragmatic Model. EPMcreate can be applied in any situation in which ideas need to be generated, e.g., at any time one might apply brainstorming (Mich et al., 2010). The feasibility of applying EPMcreate to idea generation in requirements elicitation was established by two experiments. EPMcreate demonstrated to be very effective in finding requirements that had not been known to the managers of the projects involved. Moreover, EPMcreate proved to generate more ideas and, in particular, more useful ideas than the familiar brainstorming (Mich et al., 2005). Furthermore, Mich et al. (2010) showed that EPMcreate is also effective when used by individuals.

Sakhnini et al. (2012) proposed POEPMcreate, which is an optimization of EPMcreate that requires fewer steps than EPMcreate. The effectiveness of POEPMcreate was demonstrated in two controlled experiments by comparing it to both brainstorming and EPMcreate. The results indicate that POEPMcreate is more effective, by the quantity and quality of the ideas generated, than EPMcreate, which is, in turn, more effective than brainstorming.

Karlsen et al. (2009) integrated ART-SCENE, a tool designed to discover more complete requirements with scenarios, with combinFormation, a tool that supports people in creating new ideas while finding and collecting information. As pointed out by the authors (Karlsen et al., 2009), their approach was designed to support individual creativity.

Hollis and Maiden (2013) extended Ambler's agile process with three creativity techniques: brainstorming, Partners in Creative Learning , and a new technique inspired by Hall-of-Fame. The evaluation shows that requirements generated from the extended process were rated more novel then requirements in the original product backlog.

Svensson and Taghavianfar (2015) evaluated four different creativity techniques, namely Hall of Fame, Constraint Removal, Brainstorming, and Idea Box, using creativity workshops. The creativity workshops followed the structure and the design of the creativity workshops in RESCUE (Maiden et al., 2004b). The results indicate that Brainstorming can generate by far the most ideas, while Hall of Fame generates most creative ideas. Idea Box generates the least number of ideas, and the least number of creative ideas. Finally, Hall of Fame is the technique that generates the most practical ideas (Svensson and Taghavianfar, 2015).

8 SUMMARY

In this paper, we report initial progress on a long-term project aiming at integrating collaborative games with Scrum. The proposed framework specifies a set of recommendations that aim at helping Scrum teams to choose appropriate game at a given stage of the project. The feasibility of our approach was evaluated in two commercial projects with encouraging results. We found that the adopted games: (1) made customers more willing to attend the meeting; (2) fostered stakeholders' commitment; and (3) produced better results than the standard approach. Moreover, our conversations with the project leaders indicate that they consider to use collaborative games in the future. We hope that the reported experience will also guide other practitioners to leverage collaborative games in their daily work.

Nevertheless, more research is needed to investigate the influence of collaborative games on creativity. New studies could also bring additions to the framework by exploring the application of other collaborative games and extend their usage on other aspects of software development. Finally, it is necessary to repeat the evaluation in other projects and organizations.

REFERENCES

Abt, C.C., 1970. *Serious Games*, Viking Press

Ambler, S.W., 2008. Scaling On-Site Customer. In: *Dr. Dobbs Journal*, pp. 63–66, Jan.

Baskerville, R., Myers, M.D., 2004. Special issue on action research in information systems: making IS research relevant to practice—foreward. In: *MIS Quart 28(3)*, pp. 329–335

Bjarnason, E., Unterkalmsteiner, M., Borg, M., Engström, E., 2016. A multi-case study of agile requirements engineering and the use of test cases as requirements. In: *Information and Software Technology, Vol. 77*, pp. 61-79

Boehm, B., Turner, R., 2004. *Balancing Agility and Discipline: A Guide for the Perplexed*, Addison-Wesley, Boston, MA

Bormane, L., Gržibovska, J., Bērziša, S., Grabis, J., 2016. Impact of Requirements Elicitation Processes on Success of Information System Development Projects. In: Information Technology and Management Science, vol. 19(1), pp. 57-64

Cao, L., Ramesh, B., 2008. Agile requirements engineering practices: an empirical study. In: *IEEE Softw. 25(1)*, pp. 60–67

Caroli, P., Caetano, T., 2016. *Fun Retrospectives - Activities and ideas for making agile retrospectives more engaging*, Leanpub

Carmel, E., Whitaker, R., George, J., 2016. PD and Joint Application Design: A Transatlantic Comparison. In: *Communications of the ACM, vol. 36(4)*, pp. 40–48, June

Chan, F.K.Y., Thong, J.Y.L., 2009. Acceptance of agile methodologies: A critical review and conceptual framework. In: *Decis. Support Syst. 46(4)*, pp. 803–814, March

Cohn, M., 2005. *Agile Estimating and Planning*, Addison-Wesley

Conboy, K., Wang, X., Fitzgerald, B., 2009. Creativity in Agile Systems Development: A Literature Review. In: *CreativeSME, vol. 301 of IFIP Advances in Information and Communication Technology*, pp. 122–134, Springer

Conboy, K., Coyle, S., Wang, X., Pikkarainen, M., 2010. People over process: key people challenges in agile development. In: *IEEE Software, 99*, pp. 47–57

Davis, C.J., Fuller, R.M., Tremblay, M.C., Berndt, D.J., 2006. Communication challenges in requirements elicitation and the use of the repertory GRID technique. In: *J. Comput. Inf. Syst. 47*, pp. 78–86

Davison, R.M., Martinsons, M.G., Kock, N., 2004. Principles of Canonical Action Research. In: *Information Systems Journal 14(1)*, pp. 65–86

Dawson, C., 2002. *Practical Research Methods: A User-Friendly Guide to Mastering Research Techniques and Projects*, How To Books Ltd

Derby, E., Larsen. D., 2006. *Agile Retrospectives: Making Good Teams Great*, Pragmatic Programmers

Easterbrook, S.M., Singer, J., Storey, M.A., Damian, D., 2006. Selecting Empirical Methods for Software

Engineering Research. In: *F. Shull, J. Singer and D. Sjøberg (eds) Guide to Advanced Empirical Software Engineering*, Springer

Fernandes, J., Duarte, D. Ribeiro, C., Farinha, C., Pereira, J., da Silva, M.M., 2012. iThink: A Game-Based Approach Towards Improving Collaboration and Participation in Requirement Elicitation. In: *Procedia Computer Science, vol. 15*, pp. 66–77

Faulk, S., 1997. Software Requirements: A Tutorial. In: *Thayer, R., Dorfman, M. (Eds.): Software Requirements Engineering*, IEEE Computer Society press

Garnik, I., Sikorski, M., Cockton. G., 2014. Creative sprints: an unplanned broad agile evaluation and redesign process. In: 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational (NordiCHI'14), Helsinki, Finland

Gelperin, D., 2011. Increase Requirements Understanding by Playing Cooperative Games. In: *INCOSE International Symposium*, Denver, CO

Ghanbari, H., Similä, J., Markkula, J., 2015. Utilizing online serious games to facilitate distributed requirements elicitation. In: *Journal of Systems and Softwar, vol. 109 (November)*, pp. 32–49

Gonçalves, L., Linders, B., 2014. *Getting Value out of Agile Retrospectives: A Toolbox of Retrospective Exercises*, Leanpub

Gray, D., Brown, S., Macanufo, J., 2010. *Gamestorming. A Playbook for Innowators, Rulebreakers, and Changemakers*, O'Reilly Media

Highsmith, J., Cockburn, A., 2001. Agile Software Development: The Business of Innovation. In: *IEEE Computer, vol. 34(9)*, pp. 120–122, Sep.

Hoda, R., Noble, J., Marshall, S., 2011. The impact of inadequate customer collaboration on self-organizing Agile teams. In: *Information and Software Technology 53*, pp. 521–534

Hoffmann, O., Cropley, D., Cropley, A., Nguyen, L., Swatman, P., 2005. Creativity, requirements and perspectives. In: *Australian Journal of Information Systems, vol. 13(1)*, Sep.

Hohmann, L., 2006. *Innovation Games: Creating Breakthrough Products Through Collaborative Play*, Addison-Wesley Professional

Hohmann, L., 2017. Innovation Games Website. www.innovationgames.com

Hollis, B., Maiden, N., 2013. Extending Agile Processes with Creativity Techniques. In: *IEEE Software, vol. 30(5)*, pp. 78–84

Horkoff, J., Maiden, N., 2015. Creativity and Conceptual Modeling for Requirements Engineering. In: *5th International Workshop on Creativity in Requirements Engineering*, Essen, Germany

International Institute of Business Analysis (IIBA), 2011. Agile Extension to the BABOK®Guide. Toronto, Canada

Jarzębowicz, A., Marciniak, P., 2017. A Survey on Identifying and Addressing Business Analysis Problems. In: *Foundations of Computing and Decision Sciences, Vol. 42(4)*, pp. 315–337

Jarzębowicz, A., Połocka, K., 2017. Selecting Requirements Documentation Techniques for Software Projects: a Survey Study. In: *1st International Conference on Lean and Agile Software Development*, pp. 1189–1198, http://dx.doi.org/10.15439/2017F387

Karlsen, K., Maiden, N.A.M., Kerne, A., 2009. Inventing Requirements with Creativity Support Tools. In: *15th International Working Conference, REFSQ'09*, Amsterdam, The Netherlands

Krivitsky, A., 2015. *Agile Retrospective Kickstarter*, Leanpub

Larman, C., 2003. *Agile and Iterative Development: A Manager's Guide*, Addison Wesley

Lemos, J., Alves, C., Duboc, L., Rodrigues, G., 2012. A Systematic Mapping Study on Creativity in Requirements Engineering. In: *27th ACM SAC - Requirements Engineering Track*, Riva Del Garda, Italy

Levitt, T., 1980. Marketing Success Through Differentiation – of Anything. In: *Harvard Business Review*, Jan/Feb, pp. 20–28

Maciaszek, L., 2005. *Requirements Analysis and Systems Design*, Addison-Wesley

Mahaux, M., Nguyen, L., Gotel, O., Mich, L., Mavin, A., Schmid, K., 2013. Collaborative creativity in requirements engineering: analysis and practical advice. In: *7th IEEE International Conference on Research Challenges in Information Science (RCIS)*, Paris, France

Maiden, N., Gizikis, A., Robertson, S., 2004a. Provoking creativity: imagine what your requirements could be like. In: *IEEE Software, vol. 21(5)*, pp. 68–75

Maiden, N., Manning, S., Robertson, S., Greenwood, J., 2004b. Integrating Creativity Workshops into Structured Requirements Processes. In: *5th Conference on Designing Interactive Systems: processes, practices, methods, and techniques*, Cambridge, MA

Maiden, N., Jones, S., Karlsen, I. K., Neill, R., Zachos, K., Milne, A., 2010. Requirements Engineering as Creative Problem Solving: A Research Agenda for Idea Finding. In: *18th IEEE International Conference on Requirements Engineering*, Sydney, Australia

McBreen, P., 2003. *Questioning Extreme Programming*, Addison-Wesley, Boston, MA

Mich, L., Anesi, C., Berry, D.M., 2005. Applying a pragmatics-based creativity-fostering technique to requirements elicitation. In: *Requirements Engineering, vol. 10(4)*, pp. 262–275, November

Mich, L., Berry, D.M., Alzetta, A., 2010. Individual and end-user application of the EPMcreate creativity enhancement technique to website requirements elicitation. In: *Workshop on creativity in requirements engineering at REFSQ'10*, Essen, Germany

Nerur, S., Mahapatra, R., Mangalaraj, G., 2005. Challenges of migrating to agile methodologies. In: *Commun. ACM 48*, pp. 72–78

Nguyen, L., Cybulski, J., 2008. Into the future: inspiring and stimulating users' creativity. In: *12th Pacific Asia Conference on Information Systems*, Suzhou, China

Nuseibeh, B., Easterbrook, S., 2000. Requirements Engineering: A Roadmap. In: *Conference on the Future of Software Engineering*, Limerick, Ireland

Ossowska, K., Szewc, L., Weichbroth, P., Garnik, I., Sikorski, M., 2016. Exploring an Ontological Approach for User Requirements Elicitation in the Design of Online Virtual Agents. In: *Wrycza S. (eds) Information Systems: Development, Research, Applications, Education. SIGSAND/PLAIS 2016. Lecture Notes in Business Information Processing, vol. 264*. Springer, Cham

Przybyłek, A., 2014. A Business-Oriented Approach to Requirements Elicitation. In: *9th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE'14)*, Lisbon

Przybyłek, A., Olszewski, M., 2016. Adopting collaborative games into Open Kanban. In: *2016 Federated Conference on Computer Science and Information Systems (FedCSIS'16)*, Gdansk, Poland, http://dx.doi.org/10.15439/2016F509

Przybyłek, A., Kotecka, D., 2017. Making agile retrospectives more awesome. In: *2017 Federated Conference on Computer Science and Information Systems (FedCSIS'17)*, Prague, Czech Republic, http://dx.doi.org/10.15439/2017F423

Przybyłek, A., 2017. An empirical study on the impact of AspectJ on software evolvability. In: *Empir Software Eng*, https://doi.org/10.1007/s10664-017-9580-7

Ramesh, B., Cao, L., Baskerville, R., 2010. Agile requirements engineering practices and challenges: an empirical study. In: *Inf. Syst. J., vol. 20(5)*, pp. 449–480

Redlarski, K., Weichbroth, P., 2016. Hard lessons learned: delivering usability in IT projects. In: *2016 Federated Conference on Computer Science and Information Systems (FedCSIS'16)*, Gdansk, Poland, http://dx.doi.org/10.15439/2016F20

Ribeiro, C., Farinha, C., Pereira, J., da Silva, M.M., 2014. Gamifying requirement elicitation: Practical implications and outcomes in improving stakeholders collaboration. In: *Entertainment Computing, vol. 5(1)*, pp. 335–345, December

Robertson, J., 2005. Requirements analysts must also be inventors. In: *IEEE Software vol. 22(1)*, pp. 48–50

Rodriguez, P., Markkula, J., Oivo, M., Turula, K., 2012. Survey on agile and lean usage in finnish software industry. In: *ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, Lund, Sweden

Sakhnini, V., Mich, L., Berry, D.M., 2012. The effectiveness of an optimized EPMcreate as a Creativity Enhancement Technique for Website Requirements Elicitation. In: *Requirements Engineering, Vol. 17(3)*, pp. 171–186, Sept.

Svensson, R.B., Taghavianfar, M., 2015. Selecting Creativity Techniques for Creative Requirements: An Evaluation of Four Techniques using Creativity Workshops. In: *23rd IEEE International Requirements Engineering Conference*, Ottawa, Canada

Taylor-Cummings, A., 1998. Bridging the user-IS gap: a study of major information systems projects. In: *Journal of Information Technology 13*, pp. 29–54

Trujillo, M.M., Oktaba, H., González, J.C., 2014. Improving Software Projects Inception Phase Using Games: ActiveAction Workshop. In: *9th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE'14)*, Lisbon, Portugal

VersionOne, 2017. 11th Annual State of Agile Report. https://versionone.com/pdf/VersionOne-11th-Annual-State-of-Agile-Report.pdf

Werewka, J., Spiechowicz, A., 2017. Enterprise architecture approach to Scrum processes: sprint retrospective example. In: *2017 Federated Conference on Computer Science and Information Systems (FedCSIS'17)*, Prague, Czech Republic, http://dx.doi.org/10.15439/2017F96