Journal of Universal Computer Science, vol. 22, no. 12 (2016), 1489-1508
submitted: 27/3/16, accepted: 21/11/16, appeared: 1/12/16 © J.UCS

Agile Retrospective Games for Different Team Development Phases

Miloš Jovanović
(University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia
University of the Balearic Islands, Palma, Spain
milosj@uns.ac.rs)

Antoni-Lluís Mesquida*
(University of the Balearic Islands, Palma, Spain
antoni.mesquida@uib.es)

Nikola Radaković
(University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia
nikolar@uns.ac.rs)

Antonia Mas
(University of the Balearic Islands, Palma, Spain
antonia.mas@uib.es)

Abstract: Retrospectives in Agile software development can be used for improving processes among other issues that could affect the final product quality or the working environment. Games used in retrospective meetings influence positively the project team behaviour. In this research, games from Agile retrospectives are gathered from various sources and a new classification of games is created based on the four stage group development model proposed by Tuckman. Suitable games to be used in Forming, Storming, Norming and Performing phases are presented. The results obtained in this research can help practitioners to choose the appropriate activities for process improvement and other technical or human related factors arising in the team, depending on the development stage at which their team currently is.

Keywords: Agile Software Development, Agile retrospective, Games, Process improvement
Categories: D.2.8, D.2.9

1 Introduction

Agile methods came as a response to the traditional software development and project management, which are primarily focused on identifying and documenting numerous requests that project should fulfil at the sole beginning of the project [Wiliams & Cockburn 2003]. Adaptability is the main characteristic that would describe the Agile approach (also called lean, adaptive and extreme). Agility is the ability to balance between flexibility and stability [Boehm & Turner 2005]. Agile software development changes the nature of collaboration, coordination and communication [Moe et al. 2012].

* Corresponding author

First Agile principles were introduced in the 1930 in the automotive industry [Von Rosing et al. 2015] but nowadays Agile framework may be found in many other fields such as software development, project management, Agile enterprise or service management [Jovanovic et al. 2015]. The acceptance of Agile methods in the software development industry was positively affected by the creation of Agile Manifesto in the year 2001 [Fowler & Highsmith 2001] which presents the Agile Framework encapsulating the values and principles of all Agile practices available at that time [Dingsoyr et al. 2012]. Dyba and Dingsoyr have conducted Agile software development systematic review [Dyba & Dingsoyr 2008] where main Agile development methods are summarized: crystal methodologies, DSDM, feature-driven development, lean software development, Scrum and eXtreme Programming (XP). Results from survey by VersionOne Survey in 2011 present that more than 50% surveyed participants use the Scrum method. Also, one quarter of surveyed users stated that they use custom or Scrum/XP hybrid which demonstrates the importance of hybrid methods in the IT industry.

Use of Agile methods does not necessarily mean exclusion of traditional methods [Fernandez & Fernandez 2008], and they can be used together. Hybrid approach means combining features of plan-driven and Agile process designs [Sheffield & Lemetayer 2013], two or more Agile methods combined together [Fitzgerald et al. 2006], or simply a variation of any of plan-driven and Agile methods. In highly changing business environments, initial requirements may be found inappropriate and obsolete [Highsmith 2002; Fernandez & Fernandez 2008] because requirements are changing after initiation of the project and not only at the beginning. Boehm & Turner [Boehm & Turner 2005] discuss about difficulties in merging lightweight Agile processes with existing standard industrial processes. In [Hajjdiab et al. 2012] an example of unsuccessful transition of multi project environment from waterfall towards Scrum is presented.

The nature of Agile is mostly defined by its set of core values, principles and practices [Fowler & Highsmith 2001]. The Agile manifesto's main values and 12 principles defines the Agile Framework. Therefore any method fulfilling the values and principles may be considered Agile [Fowler & Highsmith 2001]. Proposed *values* in Agile manifesto present fundamental rules in Agile software development: individual and interactions over process and tools, working software over comprehensive documentation, responding to change over following plan, keeping the process agile and keeping the process cost effective.

Agile is not a standardized process with clearly defined steps that should be pursued. One of the practices defined in manifesto refers to software process improvement initiatives: "*At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly"* [Fowler & Highsmith 2001]. The use of the Agile framework does not impose pure business process following, rather it defines basic values and rules resulting with effective communication and collaboration between team members and client [Spundak 2014]. Agile practices provide concrete actions and a set of guidelines based on Agile values and principles. Sprint retrospective is an activity devoted to software process improvement and to inspect and adapt processes in Agile framework. Anything that affects how the team creates the product is to be discussed in the retrospective meeting. During the retrospective meeting the following topics should be assessed:

what worked well, what did not work and innovative ideas that should be introduced or improved [Schwaber 2004].

The use of game elements in agile software development is recent but incorporating games in software development processes should increase the team motivation and help participants to focus on development task and define better objectives [Yilmaz & O'Connor 2016]. Several researchers have proposed game thinking approaches in software development [Cockburn 2004; Ogland 2009; Pries-Heje et al. 2004; Yilmaz & O'Connor 2016]. Games used in Agile retrospective are simplified games in terms of number of game elements, but nevertheless they increase team motivation and have a significant impact on social behaviour and team development.

A typical game consists of participants, their interactions and established rules regulating player's behaviour, a procedure or a game system where the set up "playground" is used by players to interact and achieve the game goal based on artificial conflict or outcome. Serious or applied games propose a complete game construction, and they are designed for a primary purpose other than pure entertainment [Deterding et al. 2011]. Opinions on definition of game are still not aligned but most commonly it is referred as: "*a system in which players engage in an artificial conflicts, defined by rules, that results in a quantifiable outcome"* [Salen & Zimmerman 2003].

Gamification can create seductive experiences, be persuasive and funny and engage staff with enterprise systems [Raftopoulos 2014]. In their systematic mapping of gamification in software engineering, Pedreira et al. [Pedreira et al. 2015] have presented gamification elements and mechanics found in software scientific literature: awards, point-based rewards systems, badges, quests, voting, ranking and betting. Deterding defines gamification as use of game element in non-game contexts [Deterding et al. 2011]. Utilization of game design elements and game based thinking to engage and produce motivated behaviour of participants such as cooperation or competition is referred to as gamification [Zicherman & Cunningham 2011]. Using the game design and game mechanics out of the video game industry is trend affecting the software development area among others. Term gamification is not yet precisely defined but in this article we refer to the following: Gamification is a process or transformation in which interaction patterns, game instruments, reusable games components are engaged to resolve problems in an intentional environment that is positioned within a real world settings [Yilmaz 2013].

In the year 1965 Tuckman reviewed 50 articles on group development and, as a result, proposed the four sequential stages of small-group development: *Forming, Storming, Norming and Performing* [Tuckman 1965]. Development sequence in small groups proposed by Tuckman was selected as an appropriate framework to be used in this research. Tuckman's model may be well used to investigate Agile software development, since Agile methods are people centric and participants involved are organized in small teams. Games enable us to form collective social structures and they have significant social impact [Abt 1987]. Therefore, games may be a useful tool in small group (social structure] development.

Two main contributions are presented in this study. On the one hand, literature with games used in agile software development has been gathered, analysed and classified. An integrated list of games from various sources has been defined. The

proposed listing may be useful both for practitioners who wish to implement games in software development, and for research community the integrated list of games can be further extended or reorganised based on different criteria. On the other hand, another contribution of this research is a mapping of the games with the four group development phases, thus helping practitioners to choose the appropriate games based on the current group development of their team.

This paper is organized in 7 sections. In section 2, the use of games for process improvement in agile software development is presented. In the section that follows the group development model used in the research is described. In section 4 the research methodology is detailed. After that, results are shown in section 5 and discussed and related to other literature in section 6. In the final section 7 conclusions, limitations and future work are presented.

2 Process improvement in Agile software development

Agile retrospective meeting, one of Agile ceremonies, is devoted to process improvement and other corrective activities for resolving potential issues within the team. Before organizing a retrospective meeting it is important to investigate the history and environment of team [Derby & Larsen 2006]. Agile retrospectives help teams to learn, improve and increase business value to customers and company [Kua 2013]. Even though the Agile facilitator most probably would be selected from the team members, assumptions of the moral and environment of the project team should be reviewed. It is important to gather background information in order to be capable to tailor the retrospective meeting to the needs of the team. Those parameters may be gathered at retrospective meeting itself but can be gathered prior to the meeting through other activities.

In order to prepare an effective retrospective, the following factors should be established before the beginning of a meeting [Caroli & Caetano 2015; Goncalves & Linders 2013]:

1. Learning about history and environment: team's history, general assumptions about current state and team moral should be taken into account.
2. Retrospective goal: making sense to people why to invest time and what are the expected outcomes from the meeting. Continuous improvement may be used for few iterations but otherwise it should be more concrete.
3. Duration of retrospective: four different parameters may be used to define duration: sprint iteration length, complexity [technological or organizational), team size and level of conflicts and controversy.
4. Activities (games): After the objectives of the retrospective meeting are defined and list of participants is known, activities to be conducted in the retrospective meeting may be selected. Based on the retrospective focus and participant list, activities should be chosen to help participants to engage, think, explore and think collectively. Proper selection of activities should foster equal participation, focus the conversation and encourage new perspectives.
5. Data collection: data should be gathered objectively and in time manner.

6. Agenda (structure): retrospective meeting has to be structured in order to be effective. Exact day, time and meeting venue should be defined.

The retrospective facilitator should respect the planned retrospective agenda, but his or her main responsibility is to facilitate the retrospective meeting as a whole, and to create environment in the team to reach the expected results of the retrospective meeting. If facilitator is coming from the team it is easy to get caught up in discussion. Main obligations of the facilitator are to: manage activities, group dynamics, time and himself [Derby & Larsen 2006].

Games (activities) as one of the 6 factors mentioned above affect the Agile retrospective success. Other factors are more straight-forward while choosing the right game, or set of games, to be conducted in Agile retrospective seems to be more problematic for practitioners. Therefore, the research focus in this article was on collecting, analysing and classifying games to increase the success of Agile retrospective meetings. Games can be used as an important tool to improve retrospective meetings thus resulting with better strategy for process improvements in the Agile teams.

Several researchers have proposed game thinking approaches in software development. Cockburn observed development as a game with project resources as a limitation [Cockburn 2004]. Herakovic et al. have presented a simulation of interactive push-pull lean game for production processes [Herakovic et al. 2014]. For instance, collaborative and iterative principles of software development is brought in connection with invention and communication games. Ogland used a theory of drama to propose a game to be used by software development organizations thus improving the software development process [Ogland 2009]. Software project can be viewed as a balancing game based on decision making skills of management structures in software development process [Pries-Heje et al. 2004]. Yilmaz proposed a scrumban integrated gamification approach to guide the software process improvement [Yilmaz & O'Connor 2016]. In his work, he added game elements as an addition to scrumban process in order to address the software practitioners' social concerns. Huang et. al showed that game simulations can improve product strategy and decision making in product design [Huang et al. 2014]. Raftopoulos developed a conceptual framework for the development of responsible gamified enterprise systems with four phases: discover, reframe, envision and create [Raftopoulos 2014].

Ready-made agendas for retrospective and sequence of multiple games in the same retrospective are found in reviewed literature and they can be quite useful for practitioners who want to quickly introduce games in retrospectives. Derby and Larsen have proposed sequential division of retrospective meeting in five stages: set the stage, gather data, generate insights, decide what to do and close retrospective [Derby & Larsen 2006]. The proposed grouping of retrospective games is related to organisation of each retrospective meeting and not to group development phases but it can be used as a good guide on how to start using retrospective games. Krivitsky used their sequential game grouping and defined three ready-made agendas to be used by teams [Krivitsky 2015]. In the case when team is not able the current group development phase, it might be interesting to start with one of Krivitsky's ready-made agendas. He proposed three ready-made agendas A, B, C and we refer to the agenda B as an example: check-in: mad sad glad afraid, sprint timeline, speedboat, tip the top and fun vs. use. Using more than one game in the retrospective meeting is useful

when team wants to experiment with more games in the same retrospective meeting. In our research the focus was to suggest the games to be used in retrospective meeting based on group development phase. Suggesting the combination of various games and sequence of the games in one retrospective meeting was left as a future research.

3 Tuckman's group development stages

Agile software development is people centric and project based, and project members are organised in small groups-teams. One team can be working on more projects or one project may be implemented by more teams.

In the year 1965, Tuckman reviewed 50 scientific papers with stages of group development over time [Tuckman 1965]. As a result of this research four general sequential stages of group development were proposed: *Forming, Storming, Norming* and *Performing*. Tuckman and Jensen proposed an extension to the aforementioned four stage model in the year 1977 [Tuckman & Jensen 1977]. Final stage of group development was added to the previous model, and it was named *Adjourning* (termination). In this research the final stage of group development - adjourning was not considered since at this stage since it presents the termination phase of the group, and in terms of process improving in retrospective meetings first four phases are more relevant. Nevertheless, both research manuscripts [Tuckman 1965; Tuckman & Jensen 1977] provide details of the four stage development model, and the researchers have used them as a primary source to map the games with small group development phases.

In his research Tuckman reviewed the literature dealing with developmental sequence in small groups and proposed his four stage sequential model which was validated in different context settings [Tuckman 1965]. In each phase of group development two activity types are identified: social or interpersonal activities and group-task. Therefore, Tuckman offered a conceptualization of changes in group behaviour, in both social and task realms across all group setting over time [Tuckman 1965]:

1. *Forming* is the phase of orientation, testing and dependence in the terms of social behaviour and group structure. Group members have individual objectives and they are task *oriented* in terms of group-task aspect. It is important to lay out group purpose and objectives, and set rules of behaviour in the group and expectations [Blanchard & Johnson 1982]. Also, helping individuals how they fit into group [Wetlaufer 1994].
2. *Storming* in the terms of social and interpersonal aspects, represents the phase of intragroup conflict, known for polarization around: interpersonal issues, facts, objectives, values and methods. Five stages of conflict are defined by Schmidt and Tannenbaum [Tannenbaum & Schmidt 1958]: Anticipation, conscious but unexpressed difference, discussion, open dispute and open conflict. The earlier the conflict is identified the better chance to better influence the current situation. Possible intervention measures are: focus group efforts towards building up trust and interaction, identification of protagonist and resolving out of the group and ensuring that differences are directed toward some idea and not toward individual. In terms of group-

task relation, group members begin to have emotional response to task demands.

3. *Norming* starts when resistance is replaced with in-group feeling and sense of group cohesion in terms of social and interpersonal aspects. Group standards and processes evolve and group roles are adopted. Group leader should focus on group processes development and group based big picture perspective [Wetlaufer 1994]. In this phase direct leading of group should be reduced and group leader should perform advisory role. Self-organising teams and leader acting as a facilitator rather than manager is one of key issues for Agile teams and reaching *Norming* phase of the team is important successful acceptance of Agile methods. Group members start the open discussion and exchange relevant interpretations related to tasks.
4. *Performing* is the phase where group roles become flexible and energy is focused on successful task completion. Individual group members are taking care of performance [Wetlaufer 1994] and project leader should take the supporting role. Project leaders should further encourage problem-solving approaches, and drive intrinsic motivation of individuals who desire to stretch their own limits and potential. Subjective relationship between members has already been established and they can now adopt and play roles that will enhance the task activities of the group. In task activity development this stage is identified as emergence of solutions phase.

As a summary of previously described small group development phases, the Tuckman model is concisely described. In the *Forming* phase group members are oriented on testing, dependence and orientation. *Storming* is the second phase of group development where conflict and polarization among members may be expected, and group behaviour issues will be discovered and should be resolved. In the *Norming* phase group standards and processes evolve and group roles are adopted. In the *Performing* phase roles become flexible and problem solving approach is expected. Using games in retrospectives can help in each phase of group development, thus fostering adoption of team roles, problem solving approach or resolving group behaviour issues.

Tuckman's model of small group development has been largely used in various research literature. The research on small group sequential development phases [Tuckman 1965] proposed by Tuckman is highly referenced work used as basis for specific group development models (such as groups in online environments or others) and is referred to as one of the most complete group development theory. The group of authors referred to the Tuckman's research when they investigated the relation of group development and group learning [Raes et al. 2015]. Their results show that learning processes such as sharing information, co-construction and constructive conflict occur in different group development phases. Srba and Belikova have adapted the Tuckman's small group development model for groups working in online environments - such as students enrolled in the same course [Srba & Bielikova 2014]. *Storming* and *Norming* phases where relationships and common collaboration plan development were removed due to short term orientation of the group. Jonhson et al. proposed a modification of Tuckman's four phase model for virtual learning teams. As a result of their research storming phase is removed and conflict resolution was added after *Performing* phase resulting with iterative cycle of group development

phases as follows: Forming, Norming, Performing and Conflict resolution [Johnson et al. 2002]. Group of authors presented four phases of long term virtual group development: formation, consolidation, development and closing [Gimenez et al. 2002; Daradoumis et al. 2002]. In their research of New Product Development team behaviour Dayan et al. referred to Tuckman's group development model and suggested that companies would do well to consider phases through which team progresses [Dayan et al. 2014]. Team leaders should understand team development as a continuous learning journey consisting from different steps [Griesinger & Schmitt 2016]. Collins et al. proposed a five step guide on how to implement a growth modelling approach in exploring the team dynamics [Collins et al. 2016]. Extensive collection of team data is required to follow this approach. Models proposed by Tuckman and Gersick are considered as the most complete team development theories. Tuckman's model becomes the base for many others specialized groups' lifecycle model [Srba & Bielikova 2014].

4 Methodology

The analysis of the relations between the Agile retrospective games and the group development phases of Tuckman's model was done by following an evolving strategy in which each one of the games was compared with the four group development phases proposed by Tuckman [Tuckman 1965]. This successive refinement process flow consists of three sequential stages:

4.1 Primary source selection and analysis

Experience reports and detailed explanation of games to be used in retrospective can be found in various literature. The first research stage had the aim of sharing knowledge and contrasting the different points of view of the various authors and sources presenting games used in Agile retrospectives. The result of this research step was to identify primary sources of Agile retrospective games to be used for further mapping with the group development phases proposed by Tuckman.

Firstly, the researchers were searching for the potential literature sources to be analysed. It was important to find relevant and well developed sources. Many blogs and experience reports can be found in different communities such as Agile Alliance, Scrum alliance, Project Management Institute and others. Each researcher reviewed independently the potential sources of Agile retrospective games to be used in the research. Intention of the researchers was to discover well detailed and structured resources. Therefore, internet sources such as blogs and communities were very useful to identify the primary sources of Agile retrospective games and were reviewed in the initial stage of resource search but they were not used as a source in the second stage of the research method. The sources that were lacking details of game description, practical application or had focus on the retrospective meeting agenda, rather than describing games (tools) to be used were not evaluated in the second research stage.

After reviewing the available literature on the retrospective games, 10 potential books about Agile retrospective games were found and used in the further research. It took several working sessions (joint meetings) to analyse and select the resources to

be used in the second stage of research. During each meeting two or three sources (books) of the retrospective games were analysed.

After a detailed analysis, five primary sources were selected by authors as most complete, relevant and appropriate for this research:

1. *The Retrospective Handbook* [Kua 2013]. Patrick Kua presents retrospective fundamentals, how to prepare, how to facilitate retrospective, first facilitation tips and implications for *distributed* retrospectives. Business games are divided in three groups: after the retrospective, common retrospective smell and keeping retrospectives fresh.
2. *Agile retrospective Kickstarter* [Krivitsky 2015]. Alexey Krivitsky presents cheat sheets with 16 exercises that can be combined in more than 250 variations. Games are divided in accordance with 5 stage phases proposed by Derby and Larsen [Derby & Larsen 2006]. He proposes three ready-made agendas to be used by practitioners and defines experience deducted common pitfalls and deadly sins to be avoided in retrospectives.
3. *Getting value out of Agile Retrospectives - A Toolbox of Retrospective Exercises* [Goncalves & Linders 2013]. Luis Goncalves and Ben Linders define set of retrospective exercises. Also, business value of Agile retrospective, pre requirements needed for one retrospective and potential benefits are summarized.
4. *Fun Retrospectives - Activities and ideas for making agile retrospectives more engaging* [Caroli & Caetano 2015]. Paulo Caroli and Taina Caetano [Caroli & Caetano 2015] present many activities useful for retrospective gathered from different practicioners. Activities are grouped in the following categories: energizers, check-in, building teams, looking back in retrospectives, looking ahead in retrospectives, filtering and check-out. Many games are presented with detailed guidelines and graphical presentation, thus providing valuable practical insight in retrospective gamification.
5. *Agile Retrospectives - Making Good Teams Great* [Derby & Larsen 2006]. Esther Derby and Diana Larsen present five groups of activities to be used in the same retrospective meeting at different point of time: set the stage, gather data, generate insight, decide what to do and close phase. Activities should be executed respectively starting with setting the stage and terminating with the closing phase. Retrospective facilitator should choose the appropriate games depending on the current situation of the project and team behaviour, not necessarily activities from each of the groups should be performed.

Other secondary sources were used as complementary to obtain more detailed description and practitioner experience on the games:

1. *Game storming - A Playbook for Innovators, Rulebrakers, and Changemakers* [Gray et al. 2010].
2. *Innovation Games: Creating Breakthrough Products Through Collaborative Play* [Hohmann 2006].
3. *Fifty Quick Ideas to Improve Your Retrospectives* [Roden & Williams 2015].
4. *Project Retrospectives - a handbook for team reviews* [Kerth 2013].
5. *Retrospectives for Organizational Change - An Agile Approach* [Eckstein 2014].

4.2 Creation of an integrated list of games for Agile retrospectives

In the second research stage, each game in the five selected primary sources was analysed in detail. In the former stage, the data analysis level was higher and the sources were reviewed as a whole.

As each book is organized in a different way, the first step was to compare game groupings. Depending on the book, games are divided in different groups depending on various factors. The most common factors influencing the division of games are: solution focused or goal driven games, self-managed retrospectives, refreshing retro, energisers, games fostering skillset, setting up the theme, establishing group rules or behaviour, check in games, building teams, looking back retrospectives, futurespectives, filtering, checkout and etc. [Krivitsky 2015; Gray et al. 2010; Kua 2013; Gonçalves & Linders 2013; Caroli & Caetano 2015].

Moreover, it was identified that some games are called differently in different sources, but when the game description was reviewed the similarity was noticed. In the first joint meeting of the researchers, the differences and similarities of books and games were reviewed.

During the following reviews of the research team an integrated list of games was created to be used for further research. The traceability of game description was made and high level description of the game groups proposed by each book was reviewed. For instance, the game Sailboat in the book *Getting value out of Agile Retrospectives* is named Speed boat in the book *Agile retrospective Kickstarter*. Having more descriptions of the same game was very useful and it was used as complementary in the research after the initial mapping of similar games. In the last column of Table 1 multiple sources of the game description can be traced.

4.3 Mapping of the games and group development phases

In the third stage of the research process, the aim was to identify the level of usability of each game for each group development phase proposed by Tuckman. The list of games from the five primary sources, the result from previous research stage, was used to perform a complete mapping of the games with the four phases of group development proposed by Tuckman [Tuckman 1965].

When mapping each game from the integrated list to one or more group development phase, the researchers used the detailed description of the game from one or multiple books depending on the game description availability. On one hand, books contain different game information and game grouping such as: objective of the game, activities to be followed in the game (how to do it), length of the game, graphical presentation of the game etc. On the other hand, according to Tuckman, the group development phases represent the changes in team behaviour, in both social and task aspects, across group settings throughout the whole project lifecycle.

For instance, following the Tuckman's model, in the first phase of group development called *Forming* typical behaviour of the group members would be: testing and dependence in terms of group structure, and orientation to task in terms of task activity development. Three categories were used to demonstrate the level of usability of the game in the group development phase: *best fit*, *good fit* and *possible fit*. One game that is identified as a *best fit* in one group development phase does not necessarily mean that it cannot be used in another group development phase, it means

that based on the analysis of authors the game is a best fit to be used in specific group development phase. When the game description was a clear fit to certain group development phase, it was marked as a *best fit*. The other two categories (*good fit* and *possible fit*) came as a result of authors consensus when the suitability of the game was not clear. Therefore, as a result of the third research stage the authors created the Table 1 with three levels of "fitting" for each group development phase.

5 Results

Table 1 presents a list of possible games to be used during an Agile retrospective meeting. Not all the games from the five primary sources are contained in the table. If in the opinion of the researchers, it was not clear to which group the game should be related, the game was not presented in the final table. Gathering games used in the retrospective was not the primary objective of this research study and, therefore, only the games that could be related to the group development phase(s) of Tuckman's model [Tuckman 1965] are presented.

Table 1 should help practitioners to choose games for retrospective meetings depending on the current stage of the group development. The table consists of six columns. First column shows the name of the games in the alphabetical order. Second, third, fourth and fifth columns demonstrate which games could be used in each of the group development phases proposed by Tuckman. Last column, named *Source*, details literature source where detailed explanation of each game may be found: S1 - [Kua 2013], S2 - [Krivitsky 2015], S3 - [Gonçalves & Linders 2013], S4 - [Caroli & Caetano 2015] and S5 - [Derby & Larsen 2006].

The games presented may be used in different phases of group development. In the Table 1, three levels of games "fitting" to group development phase is presented: the *best fit* games with symbol ●●●, *good fit* with symbol ●● and *possible fit* with symbol ●.

Games	Forming	Storming	Norming	Performing	Source
360 degrees appreciation	●	●●●	●●●	●●●	S4
4L	●	●●●	●●●	●●●	S4
5WHY	●●	●●●	●●	●●	S3
Anchors and engine	●	●●●	●●●	●●●	S1,S4
Aprreciative inquiry retros	●	●●	●●●	●●●	S1
Asking questions	●●	●●●	●●●	●●	S3
Back to back	●●●	●●	●	●	S4
Balloon Battle	●●●	●●	●	●	S4
Car brand	●●	●●●	●●	●	S3
Clear trade off sliders	●●●	●●●	●●●	●●	S4
Collaborative product vision	●●●	●●●	●●●	●●	S4
Complex pieces	●●●	●	●●	●●	S4
Constellations	●	●●	●●●	●●●	S2,S3
Continious retrospectives	●●	●●●	●●●	●●	S1
Creating safety	●●	●●●	●●	●●	S4
DAKI	●●	●●●	●●●	●●●	S4
Defining Nirvana	●●●	●●●	●●●	●●	S4
Defining team principles	●●●	●●●	●●●	●	S4

Games	Forming	Storming	Norming	Performing	Source
Defining Team vision statement	●●●	●●●	●●●	●●	S4
Delegation map	●●●	●●●	●●●	●●	S4
Dialogue sheets	●●	●●●	●●●	●	S1
Empathy snap on big moment	●●	●●●	●●●	●●●	S4
Explorer, shopper, vacationer, prisoner	●●	●●●	●●●	●●	S4, S5
find your pair	●●●	●●	●	●	S4
Fishbowl discussion	●●	●●	●●●	●●	S1
FLAP	●	●●●	●●●	●●●	S4
FMEA	●	●●●	●●●	●●●	S4
Focus on/Focus off	●●	●●	●●●	●●●	S5
Following up on action items	●	●●●	●●●	●●●	S4
Forming triangles	●●●	●●	●	●	S4
Fun vs. Use	●●●	●●●	●●●	●●	S2
Futurespective	●	●●	●●●	●●	S1
General behaviour activity	●●●	●●●	●●●	●●	S4
Geographic location	●●●	●	●	●	S4
Ground rules	●●●	●●●	●●●	●	S4
Happines index	●●	●●●	●●●	●●●	S3
Happines radar	●●	●●●	●●●	●●	S4
High performance tree	●●●	●●●	●●●	●●	S3
Hot air ballon	●●	●●●	●●●	●●●	S1,S4
KALM	●●	●●●	●●●	●●●	S4
Known issues	●	●●●	●●●	●●●	S4
Last retro follow up	●	●●●	●●●	●●	S2
Like to like	●●	●●●	●●●	●●	S5
Locate strengths	●●	●●●	●●●	●●●	S5
Mad, sad, glad, afraid	●●●	●●●	●●●	●●	S2, S5
More, less, keep, stop, start	●●	●●●	●●●	●●	S2
One two ping four Pong	●●●	●●	●●	●	S4
One word retro	●●	●●●	●●●	●●	S2, S3, S4
Open the box	●	●●●	●●●	●●●	S4
Peaks and Valley timeline	●	●●	●●●	●●●	S4
Peer introduction games	●●●	●●●	●●●	●●	S4
PMI	●●	●●●	●●●	●●●	S4
Problem, experiment, who,what, when	●●	●●●	●●●	●●	S2,S4
Proud, thankful, learned	●	●●●	●●●	●●	S2
Punctual Paulo	●●●	●●	●	●	S4
Repeat/avoid	●	●●●	●●●	●●●	S4
Retro of retro	●●●	●●●	●●	●●	S3
Role expectations matrix	●●●	●●●	●●●	●●	S4
Safety check	●●	●●●	●●●	●	S4
Satisfaction histogram	●●	●●●	●●●	●●	S5
Solution focused/goal driven	●●	●	●●●	●●	S1
Speed car with abys	●●	●●●	●●●	●●●	S4
Speed (or sail) boat	●●●	●●●	●●●	●●	S2,S3
Sprint perfection game	●	●●●	●●●	●●●	S2

Games	Forming	Storming	Norming	Performing	Source
Sprint timeline	●●	●●●	●●●	●●●	S2
Starfish	●●	●●●	●●●	●●●	S3,S4
Strength based retro	●●	●●	●●●	●●●	S3
Team assessment survey	●	●●	●●●	●●●	S3
Team radar	●●	●●	●●●	●●●	S5
That guy and this guy	●●●	●●●	●●●	●	S4
The story of a story	●●	●●●	●●●	●●●	S4
Thums up, down, new ideas and acknowledgment	●●	●●●	●●●	●●●	S4
Timeline driven by feelings	●	●●●	●●●	●●●	S4, S5
Tip the top/break out	●	●●●	●●●	●●	S2
Token of appreciation	●	●●●	●●●	●●●	S4
Train	●	●●●	●●●	●●	S1
Triple nickles	●	●●	●●●	●	S5
Untangle yourselves	●●●	●●	●●	●	S4
Value stream mapping	●●	●●	●●●	●●●	S3
Visual phone	●●●	●●	●●	●	S4
Weather forecast	●●	●●	●●●	●●●	S1
Who am I?	●●●	●●	●●●	●	S1
Working well needs fixing	●●	●●●	●●●	●●	S2
WWW	●●	●●●	●●●	●●●	S4
Zip Zap Zoom	●●●	●●	●	●	S4

Table 1: Games used in group development phases

6 Discussion

Agile projects are implemented by small project teams and small group development should be considered as an important factor when planning the retrospective meeting and choosing the right games to be conducted. In this section, the findings presented in Table 1 are discussed. Moreover, the grouping of games by other authors is reviewed and the obtained results are associated with related research.

1. *Forming* phase of group development is the phase of orientation and conversion of individual objectives towards group objectives [Tuckman 1965; Tuckman & Jensen 1977]. This change is especially important for teams transforming from traditional organisational culture towards Agile way of working. The group members should change their individual objectives and task responsibility towards group objectives and group responsibility [Qumer & Henderson-Sellers 2008]. In this phase, people relations, expectations and behavioural ground rules are set. Therefore, this first phase includes games such as energizers, ice-breakers, warm up and building teams. In this phase there is not much data existing that could be gathered or analysed, since team members should establish basic values to be used in the future, get to know each other on personal and technical level and start to be productive as soon as possible.
2. *Storming* group development phase is known by polarization and conflicts related to interpersonal issues and personal/team objectives and values. The

conflicts should be resolved as soon as possible, especially before becoming an open conflict. To be able to resolve conflicts, first they have to be identified. Therefore, games in this group development phase, among others, include: check in, self-managed sheet retrospectives, building teams, looking back retrospectives and looking ahead (futurespectives). In this phase, it is important to discover potential conflicts among team members, check if values and ground rules are properly set in previous stage, check if group roles are clear and well defined and provide team building activities. Agile facilitator should foster discussion about personal issues, group values, resolve conflicts, gather data from previous activities and analyse obstacles.

3. *Norming* group development phase starts with termination of resistance and polarization in the group. Group standards and processes evolve and team roles are adopted. Software process improvement should be conducted in each retrospective meeting. At this point of time phase process are well established in the group and further improvements are welcomed. Therefore, games in this development phase, among others, include: check in, self-managed sheet retrospectives, building teams, looking back retrospectives, looking ahead (futurespectives), everything goes well and fostering skillset. At this stage, significant data may be obtained and analysed. New conflicts will continue to arise but majority of tension should have already been resolved. So, the facilitator should focus on gathering and analysing data, team skillset development, refreshing retrospective and using games with specific focus on certain details.
4. *Performing* is a group development phase when group roles become flexible and project leader takes the supporting role. This is the very productive phase of group development where rules and roles are clearly defined and established. Group members are motivated to propose improvements and further develop their skills and relations among them. Therefore, games in this development phase, among others, include: everything goes well, fostering skillset, looking back retrospectives and retrospectives of retrospectives.

The summary of relations identified between the group development phases and related literature presented in previous 4 paragraphs:

1. *Forming:* Energizers, ice-breakers, warm up and building teams.
2. *Storming:* check in, self-managed sheet retrospectives, building teams, looking back retrospectives and looking ahead (futurespectives).
3. *Norming:* check in, self-managed sheet retrospectives, building teams, looking back retrospectives, looking ahead (futurespectives), everything goes well and fostering skillset.
4. *Performing:* everything goes well, fostering skillset, looking back retrospectives and retrospectives of retrospectives.

Each of the four small group development phases proposed by Tuckman embodies different characteristics of member behaviour, and researchers have reviewed each game presented in Table 1 to propose the “fitting” level of the games with group development phase. Certain relations of group development phase and game groupings proposed by authors of primary source literature may be outlined. For

instance, energizers and warm up games fit well to the *Forming* group phase and fostering skillset group of games fits well to the *Norming* group development phase. In the book Fun Retrospectives, the energizer games are described as "optional activities that take place to warm up the team and promote group interaction. Good starter for any team meeting. It is extra valuable for early stages of team building."[Caroli & Caetano 2015]. It is explicitly stated by the book authors that energizer games are very valuable for early stages of team building. Therefore, it is not a surprise that games from this group were marked with perfect fit (●●●) to the *Forming* group development phase in Table 1. Energizer games may be used in other group development phases but they are most useful for the *Forming* group phase.

7 Conclusion and future work

Games may be used in retrospective meetings in Agile software development for improving processes. In this research, games used in Agile retrospectives were gathered from various sources and a new classification of games was created based on the four stage small group development model proposed by Tuckman: *Forming, Storming, Norming and Performing*. Tuckman's phases were very well described in different literature and a good model to apply in this research.

It can be concluded that games encountered to be used in retrospectives consist of quite simple gamification elements. In majority of games voting and ranking can be found but other elements such as point rewards systems or badges are not present.

Software development teams working in an Agile environment are used to iterative development and constant changes. Games may be very well used in such environments. Most common place for games in Agile software development would be Agile retrospective but games may be used in other ceremonies as well (iteration planning and review meetings).

One of the contributions for academy and industry is the list of literature and games that could be used in retrospective meetings. Firstly, ten books were identified as the most relevant, presenting the repository of game descriptions, experience reports and other relevant factors for the use of the games in retrospectives. After analysis, the researchers grouped the literature in primary and secondary sources, thus pointing out the hierarchy and a starting point for potential readers. The presented literature and list of games may be used regardless of defining the group development phase.

Another contribution of this research is the integrated list of games presented in Table 1, which relates the suitable games to be chosen for a retrospective depending on the group development phase. The results obtained in this research can help practitioners to decide the appropriate activities for process improvement and to address other technical and human related issues in the team. Depending on the team background and current situation of the project different games may be chosen, leader and team as a whole should choose the adequate set of the games.

The Tuckman's *Adjourning* group development phase was not considered in this research since it is related to the separation and termination of the team. The first four phases of small group development were considered as the most relevant for process

improvement initiatives, but the games for the *Adjourning* phase may be investigated as a future work.

In this research, retrospective games were grouped in relation to the team development phases but in the future work the games may be grouped and observed from the perspective of the game objectives. Instead of relating the games to the group development phases, they could be grouped in terms of objectives that games should achieve. For instance, games could be split in the following grouping: people (human related aspects), tool (technical aspect), process and others. Games could be grouped in the proposed order depending on the game objective, that is, which aspect should be addressed by conducting the game.

Derby and Larson presented the agenda with five type of games that could be used sequentially in the same retrospective meeting: set the stage, gather data, generate insights, decide what to do and close the retrospective [Derby & Larsen 2006]. The proposed agenda may be useful for other meetings, apart from retrospective. Adding more games to the proposed grouping may be performed as a future work. Also, suggesting the short list of games and ready-made agendas would be useful for practitioners to quickly start using the games in their daily activities. Therefore, proposing the ordering of multiple games to be used in same retrospective and ready-made agendas may be interesting for practitioners. The researchers have started a pilot project in a company which should result in the future with multiple game groupings and ready-made agendas that could be used as a quick guide for the companies willing to use games.

Several limitations to the proposed research were identified. Team members may not be capable to determine in which phase of group development they currently are. This may be a result of a limited time or not understanding the four phase group development model proposed by Tuckman. Identifying and relating situational factors to the games may help practitioners on which game to use. Certain situational factors may be related to the games but also, group of situational factors used to identify the development phase may be identified. For instance, situational factors affecting the software development process proposed by Clarke [Clarke & O'Connor 2012] may be used to address this issue.

Another limitation to this research is the availability of game descriptions and their applicability in practice. The authors have identified 10 books as the most complete and relevant literature. Nevertheless, gamification and games used in retrospective are emerging areas and some inputs may have been left out in this research. Many information may be found on blogs and Internet, so to some extent this research study is limited in a sense that additional "informal" sources of games were not involved in this study. The intention of the researchers was to include the most detailed formal literature such as books or research articles.

Acknowledgements

This work has been supported by the Spanish Ministry of Science and Technology with ERDF funds under grants TIN2016-76956-C3-3-R and TIN2013-46928-C3-2-R and by Erasmus Mundus Euroweb+ project.

References

[Abt 1987] Abt, C.C., 1987. Serious Games, University press of America.

[Blanchard & Johnson 1982] Blanchard, K. & Johnson, B., 1982. The One Minute Manager, New York: A Berkley Book.

[Boehm & Turner 2005] Boehm, B. & Turner, R., 2005. Management Challenges to implementing agile processes in traditional software development organizations. IEEE Software, 22(5), pp.30–39.

[Caroli & Caetano 2015] Caroli, P. & Caetano, T., 2015. Fun Retrospectives - Activities and ideas for making agile retrospectives more engaging, Leanpub.

[Clarke & O'Connor 2012] Clarke, P. & O'Connor, R. V., 2012. The situational factors that affect the software development process: Towards a comprehensive reference framework. Information and Software Technology, 54(5), pp.433–447.

[Cockburn 2004] Cockburn, A., 2004. The End of Software Engineering and the Start of Economic-Cooperative Gaming. , 1(1), pp.1–32.

[Collins et al. 2016] Collins, C.G. et al., 2016. Unpacking team dynamics with growth modeling: An approach to test, refine, and integrate theory. Organizational Psychology Review, 6(1), pp.63–91.

[Daradoumis et al. 2002] Daradoumis, T. et al., 2002. Supporting the composition of effective virtual groups for collaborative learning. International Conference on Computers in Education, 2002. Proceedings., 1(1), pp.332–336.

[Dayan et al. 2014] Dayan, M., Ozer, M. & Almazrouei, H., 2014. The role of functional and demographic diversity on new product creativity and the moderating impact of project uncertainty. Industrial Marketing Management. Available at: http://dx.doi.org/10.1016/j.indmarman.2016.04.016.

[Derby & Larsen 2006] Derby, E. & Larsen, D., 2006. Agile retrospectives - Making Good Teams Great, Dallas, Texas: The Pragmatic Bookshelf.

[Deterding et al. 2011] Deterding, S. et al., 2011. Gamification. using game-design elements in non-gaming contexts. Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA '11, p.2425. Available at: http://portal.acm.org/citation.cfm?doid=1979742.1979575.

[Dingsoyr et al. 2012] Dingsoyr, T. et al., 2012. A decade of agile methodologies: Towards explaining agile software development. Journal of Systems and Software, 85(6), pp.1213–1221.

[Dyba & Dingsoyr 2008] Dybå, T. & Dingsøyr, T., 2008. Empirical studies of agile software development: A systematic review. Information and Software Technology, 50(9–10), pp.833–859.

[Eckstein 2014] Eckstein, J., 2014. Retrospectives for Organizational Change - An Agile Approach, Leanpub.

[Fernandez & Fernandez 2008] Fernandez, D.J. & Fernandez, J.D., 2008. Agile project management : Agilism versus traditional approaches. The Journal of Computer Information Systems, 49(2), pp.10–17.

[Fitzgerald et al. 2006] Fitzgerald, B., Hartnett, G. & Conboy, K., 2006. Customising agile methods to software practices at Intel Shannon. European Journal of Information Systems, 15(2), pp.200–213.

[Fowler & Highsmith 2001] Fowler, M. & Highsmith, J., 2001. The agile manifesto. Software Development, 9(August), pp.28–35.

[Gimenez et al. 2002] Gimenez, F., Guitert, M. & Daradoumis, A., 2002. Cooperative learning & working in a virtual environment: A case study. Proc. of the 2nd. Available at: http://www.uoc.edu/in3/grupsrecerca/ED-MEDIA_Brief_paper.doc.

[Goncalves & Linders 2013] Gonçalves, L. & Linders, B., 2013. Getting value out of Agile retrospectives. A toolbox of retrospective exercises,

[Gray et al. 2010] Gray, D., Brown, S. & Macanufo, J., 2010. Game storming - A playbook for Innovators, Rulebreakers, and Changemakers, O'Reilly Media.

[Griesinger & Schmitt 2016] Griesinger, A. & Schmitt, T., 2016. Training and Qualification: Recommendations for Virtual Team Development. In Handbook of Human Resource Management. Springer-Verlag Berlin Heidelberg, pp. 395–404.

[Hajjdiab et al. 2012] Hajjdiab, H., Taleb, A.S. & Ali, J., 2012. An industrial case study for Scrum adoption. Journal of Software, 7(1), pp.237–242.

[Herakovic et al. 2014] Herakovic, N., Metlikovic, P. & Debevec, M., 2014. Motivational Lean Game to Support Decision between Push and Pull Production Strategy. International Journal of Simulation Modelling, 13, pp.433–446.

[Highsmith 2002] Highsmith, J., 2002. Agile Software Development Ecosystem, Addison Wesley.

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

[Huang et al. 2014] Huang, Y.-M., Liu, L. & Qi, E.-S., 2014. Simulation on the Complementary Product Strategy Based on the Cournot-Bertrand Mixed Game Model. International Journal of Simulation Modelling, 13(4), pp.497–509.

[Johnson et al. 2002] Johnson, S.D. et al., 2002. Team development and group processes of virtual learning teams. Computers & Education, 39(4), pp.379–393. Available at: http://linkinghub.elsevier.com/retrieve/pii/S036013150200074X.

[Jovanovic et al. 2015] Jovanovic, M. et al., 2015. The Agile approach in industrial and software engineering project management. Istrazivanja i projektovanja za privredu, 13(4), pp.213–216. Available at: http://scindeks.ceon.rs/Article.aspx?artid=1451-41171504213J.

[Kerth 2013] Kerth, N., 2013. Project Retrospectives - a handbook for team reviews, New York: Dorset House Publishing.

[Krivitsky 2015] Krivitsky, A., 2015. Agile Retrospective Kickstarter, Leanpub.

[Kua 2013] Kua, P., 2013. The Retrospective Handbook, Leanpub.

[Moe et al. 2012] Moe, N.B., Aurum, A. & Dybå, T., 2012. Challenges of shared decision-making: A multiple case study of agile software development. Information and Software Technology, 54(8), pp.853–865. Available at: http://dx.doi.org/10.1016/j.infsof.2011.11.006.

[Ogland 2009] Ogland, P., 2009. The Game of Software Process Improvement: Some Reflections on Players, Strategies and Payoff. In Norsk konferanse for organisasjoners bruk av informasjonsteknologi (NOKOBIT-16). Trodenheim, Norway, pp. 23–25.

[Pedreira et al. 2015] Pedreira, O. et al., 2015. Gamification in software engineering - A systematic mapping. Information and Software Technology, 57(1), pp.157–168. Available at: http://dx.doi.org/10.1016/j.infsof.2014.08.007.

[Pries Heje et al. 2004] Pries-Heje, J. et al., 2004. The high speed balancing game. Scandinavian Journal of Information Systems, 16, pp.11–54. Available at: http://www.researchgate.net/publication/220531857_The_High_Speed_Balancing_Game/file/32bfe511010e253b5b.pdf.

[Qumer & Henderson-Sellers 2008] Qumer, A. & Henderson-Sellers, B., 2008. A framework to support the evaluation, adoption and improvement of agile methods in practice. Journal of Systems and Software, 81(11), pp.1899–1919.

[Raes et al. 2015] Raes, E. et al., 2015. An Exploratory Study of Group Development and Team Learning. Human Resource Development Quarterly, 26(1), pp.5–29. Available at: http://www.robots.ox.ac.uk/~vgg/rg/papers/efficientmatch.pdf.

[Raftopoulos 2014] Raftopoulos, M., 2014. Towards gamification transparency: A conceptual framework for the development of responsible gamified enterprise systems. Journal of Gaming & Virtual Worlds, 6(2), pp.159–178.

[Roden & Williams 2015] Roden, T. & Williams, B., 2015. Fifty Quick Ideas to Improve Your Retrospectives, Leanpub.

[Von Rosing et al. 2015] Von Rosing, M., von Scheel, J. & Gill, A.Q., 2015. Applying Agile Principles to BPM, Elsevier Inc. Available at: http://linkinghub.elsevier.com/retrieve/pii/B9780127999593000276.

[Salen & Zimmerman 2003] Salen, K. & Zimmerman, E., 2003. Rules of Play: Game Design Fundamentals,

[Schwaber 2004] Schwaber, K., 2004. Agile Project Management with Scrum, Microsoft press. Available at: http://www.amazon.com/dp/073561993X.

[Sheffield & Lemetayer 2013] Sheffield, J. & Lemétayer, J., 2013. Factors associated with the software development agility of successful projects. International Journal of Project Management, 31(3), pp.459–472. Available at: http://dx.doi.org/10.1016/j.ijproman.2012.09.011.

[Spundak 2014] Špundak, M., 2014. Mixed Agile/Traditional Project Management Methodology – Reality or Illusion? Procedia - Social and Behavioral Sciences, 119, pp.939–948.

[Srba & Bielikova 2014] Srba, I. & Bielikova, M., 2014. Dynamic Group Formation as an Approach to Collaborative Learning Support. IEEE Transactions on Learning Technologies, PP(99), pp.173–186. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6963449.

[Tannenbaum & Schmidt 1958] Tannenbaum, R. & Schmidt, W.H., 1958. How to Choose a Leadership Pattern. Harvard Business Review, 36(2), pp.95–101.

[Tuckman 1965] Tuckman, B.W., 1965. DEVELOPMENTAL SEQUENCE IN SMALL GROUPS 1. , 63(6), pp.384–399.

[Tuckman & Jensen 1977] Tuckman, B.W. & Jensen, M.A.C., 1977. Stages of small-group development revisited. Group & Organisation Management, 2(4), pp.419–427.

[Wetlaufer 1994] Wetlaufer, S., 1994. The Team That Wasn't. Harvard Business Review, 72(6), pp.22–26.

[Wiliams & Cockburn 2003] Wiliams, L. & Cockburn, A., 2003. Agile Software Development: It's about Feedback and change. IEEE computer society, 36(6), pp.39–43.

[Yilmaz 2013] Yilmaz, M., 2013. A Software Process Engineering Approach to Understanding Software Productivity and Team Personality Characteristics.

[Yilmaz & O'Connor 2016] Yilmaz, M. & O'Connor, R. V, 2016. A Scrumban integrated gamification approach to guide software process improvement: a Turkish case study. Tehnicki vjesnik, 23(1), pp.237–245.

[Zicherman & Cunningham 2011] Zicherman, G. & Cunningham, C., 2011. Gamification by Design - Implementing Game Mechanics in Web and Mobile Apps, O'Reilly Media.