A Gamified Requirements Inspection Process for Goal Models

João Pimentel
Universidade Federal Rural de Pernambuco
Cabo de Santo Agostinho, Brazil
joao.hcpimentel@ufrpe.br

Emanuel Santos
Universidade Federal de Pernambuco
Recife, Brazil
emanuel.batista@gmail.com

Tarcisio Pereira
Universidade Federal de Pernambuco
Recife, Brazil
tcp@cin.ufpe.br

Daniel Ferreira
Universidade de Pernambuco
Recife, Brazil
dfs3@ecomp.poli.br

Jaelson Castro
Universidade Federal de Pernambuco
Recife, Brazil
jbc@cin.ufpe.br

ABSTRACT

THIS IS A PRE-PRINT COPY

Despite its potential benefits, requirements inspection is an often overlooked activity. When it is performed, it is usually in an unstructured, *ad hoc* fashion. This is particularly the case for requirements documented as i^* goal models, considering that there is no inspection process designed specifically to address this kind of models. In this paper, we propose a gamified requirements inspection process that goes beyond the straightforward application of game mechanics. The Ring-i process (Requirements Inspection Gamified process for i^* Models) allows stakeholders to verify i^* models together, in a playful environment. Empirical evaluation with students in a Requirements Engineering course provided indications that the process is not only useful but also fun and easy to use.

CCS CONCEPTS

• **Software and its engineering → Specification languages**; **Requirements analysis**; **Process validation**; Software development methods;

KEYWORDS

Requirements Inspection; Gamification; Goal Modeling; Social Modeling; Verification and Validation; Requirements Engineering; Software engineering

ACM Reference Format:
João Pimentel, Emanuel Santos, Tarcisio Pereira, Daniel Ferreira, and Jaelson Castro. 2018. A Gamified Requirements Inspection Process for Goal Models. In *SAC 2018: SAC 2018: Symposium on Applied Computing , April 9–13, 2018, Pau, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3167132.3167272

1 INTRODUCTION

In 1976, Bell and Thayer concluded that requirements specifications are often incorrect, ambiguous, inconsistent, or incomplete, calling for methodologies to support the verification and validation of software requirements [1].

In 1991, Sakthivel [16] analyzed different requirements verification techniques regarding their capability to detect requirements errors, concluding that reviews (in the form of walkthroughs or inspections) are the most capable technique for uncovering different kinds of errors. Nonetheless, it is noted that "their capability is contingent upon the efficiency, knowledge, and skills of the reviewers. These techniques are ad hoc and time consuming." [16]

In 2015, Kassab [9] reported on a survey of 247 practitioners, in which 55% of participants declared to perform some form of requirements review. From these, the most adopted techniques were (from more to less adopted): *ad hoc* walkthrough, checklist, formal walkthrough, and scenario-based. Thus, even though there are structured ways to perform requirements review (such as checklists and formal walkthrough), they are often not adopted.

Gamification has been successfully applied in recent years to both existing and new processes, affecting enjoyment, engagement, satisfaction, and motivation of their adopters [6]. In particular, gamification has already been applied in multiple software engineering processes and activities [13].

In this paper, we propose a process to guide the inspection of a particular kind of requirements artifact: i^* models [21]. The Ring-i process (Requirements Inspection Gamified process for i^* Models) provides a structured way to inspect i^* models in a group setting, facilitating the inclusion of stakeholders that are not familiar with the i^* modeling notation (such as clients and users). Unlike other gamified requirements engineering approaches that focus on game mechanics such as points, badges, and avatars (e.g., [3, 10, 11, 14]), the Ring-i process is based on the essential traits of games: goal, rules, feedback system, and voluntary participation [12].

The remainder of this paper is structured as follows. In Section 2 we present an overview of the i^* framework, which constitutes the baseline of our work. Section 3 describes the Requirements Inspection Gamified process for i^* Models (Ring-i), which is the main contribution of this paper. Section 4 presents an early evaluation of the proposal, which is followed in Section 5 by a fully-fledged empirical evaluation. The results of the empirical evaluation are presented in Section 6. Related work is discussed in Section 7, with concluding remarks and future work presented in Section 8.

2 BACKGROUND - THE I* FRAMEWORK

The i^* framework [21] has been widely adopted by the requirements engineering research community [8] due to its capability not only of expressing the goals and functions (tasks) of a system, but also the non-functional requirements (softgoals), the resources required to perform a function, and the relationships (dependencies) between different actors.

Figure 1 presents an example of an i^* model. The goals represent the intentions, needs or objectives of the actors. Softgoals are a particular kind of goal of subjective nature – they are not measured in concrete terms, but are generally used to describe the actors' desires related to quality attributes of their goals. The tasks represent a way to perform some activity. The resources represent data, information or a physical resource that an actor may provide or receive. The relationship between actors is expressed through dependency links – a *depender* needs a *dependee* for a *dependum* (which, itself, is a goal, softgoal, task, or resource). Within an actor, the elements are linked together through task-decompositions, means-end and contribution links. The means-end links define which alternative tasks (means) may be performed in order to achieve a given goal (end) (e.g., *Task T1* is a possible means to achieve *Goal G1*). The task-decomposition links describe what should be done to perform a certain task (e.g., *Task T1* is decomposed onto *Task T2* and *Task T3*). Finally, the contributions links suggest how a task can contribute (positively or negatively) to satisfy a softgoal, e.g., *Task T2* contributes negatively to (*Hurt*) *Softgoal S1*.

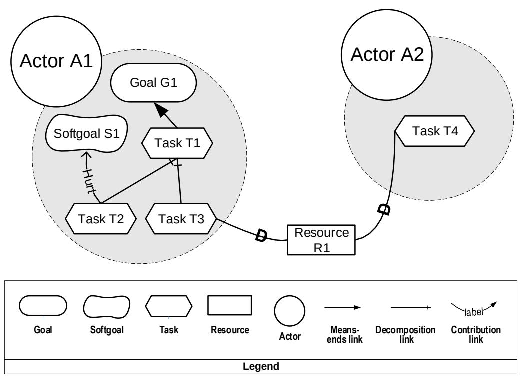

Figure 1: Example of an i^* model to illustrate its main concepts

3 THE RING-I PROCESS

In the Requirements Inspection Gamified process for i^* models (Ring-i), the i^* model to be inspected is used as a board in a board game, on which the stakeholders move their tokens over. As their tokens stop over a given element, the stakeholders draw Inspection Cards prompting them to analyze a certain aspect of the model, such as repetition of elements, scope, syntax, and textual content. Potential issues identified throughout the session are compiled into a list, which is later used to update the model.

Figure 2 shows the typical setup for a Ring-i session, with Inspection Cards, rulebook, printed model, tokens, and dice.

Figure 3 presents an overview of the Ring-i process in BPMN (Business Process Model and Notation). Its input is the i^* model to be inspected. In the Model Preparation task, the elements of the model are annotated with numbered circles, which indicate the order on which the model will be traversed. These numbers are assigned based on proximity, making a path from a start point to a finish point. This model, edited to look like a board in a board game, is the input to the Model Inspection sub-process. The output

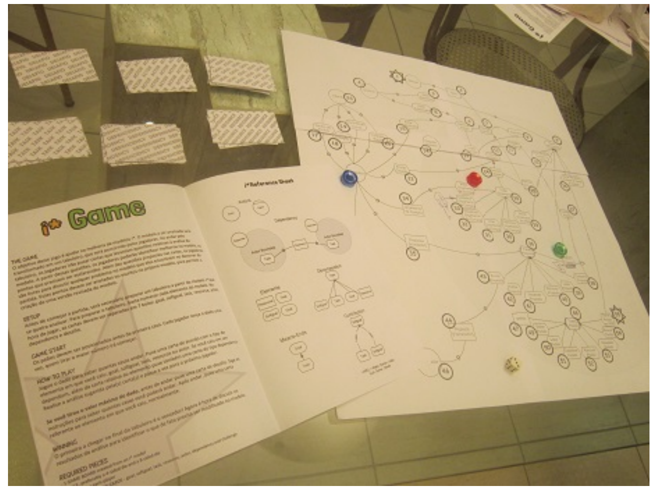

Figure 2: A Ring-i session setup: model/board, tokens, dice, rules, and cards

of this sub-process is a list of issues, to be addressed in the Model Update task, resulting in a revised i^* model. If further inspection is deemed necessary, the process begins anew.

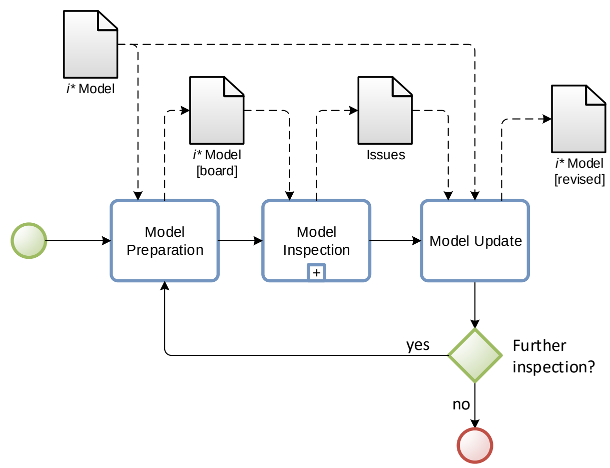

Figure 3: Overview of the Ring-i process

3.1 The Rules

Each person participating in the inspection (from here on referred to as *players*) is represented by a token on the board. The base mechanic of the actual Model Inspection is as follows: In every turn, each player rolls a die and moves her/his token through the board, which is the i^* model under inspection. Then, she/he will draw an Inspection Card, which presents inspection prompts.

As the player's token moves through the model, Inspection Cards are drawn according to the type of element on which the token is currently stopped: actor, its internal elements (goal, softgoal, task, resource), or dependency. These cards contain prompts, such as an action to be performed or questions to be answered. The player then performs the action and passes the turn to the next player. The session proceeds until a player reaches the last element of the board – that player is the winner. Figure 2 shows the elements of the i^* game: the model/board, the rulebook, the inspection cards, tokens, and dice.

Throughout the inspection session, players identify potential changes to be performed in the model, such as fixes and improvements, as well as concerns that must be further clarified with other stakeholders. In order to record these results, the players may annotate them in the model itself or compile a list of issues/changes.

In the end, the players will have inspected several elements of the model, will have identified changes to be applied in the model, and will have identified some concerns that should be clarified. According to the amount of changes required in the model, it may be fruitful to perform another round of inspection once a new version of the model is created.

3.2 Inspection Cards

The analysis of the model results from the prompts contained in the Inspection Cards, which are mostly related to a given kind of element (e.g., "What do you need to know in order to perform this task? This may be modeled as a resource?"). These cards are separated in seven stacks: actor cards, goal cards, softgoal cards, task cards, resource cards, dependency cards, and challenge cards. Thus, if the token of a player is over a goal within the boundary of an actor, a card from the respective stack will be drawn. If the token is over a goal that is part of a dependency, instead, two cards will be drawn: one from the stack of goal cards, and another from the stack of dependency cards.

Even though the cards are suited to specific element types, some cards require the player to analyze a broader set of element, such as: "The tasks linked to this goal are alternatives, or more than one of them is required in order to achieve the goal?", and "What are the main goals of this actor?" Based on these actions, the player will consider whether some modification is required, as well as what would be the modification. This discussion about the modification can involve all the players, not only the player that has drawn the card. Furthermore, when analyzing the answer to the prompt, the players are free to identify other concerns as well.

Besides these prompts, the set of Inspection Cards also contain cards related to creativity and to movement on the board. The creativity cards were created in order to break a bit the intensive analytical thinking required for the inspection, by suggesting actions related to drawing, mime, music, and so on. These cards contribute to the inspection process in three ways: by stimulating the creativity of the players; by letting the players rest their analytical function without interrupting the game; and by adding fun.

The movement cards just make the player move their tokens, to either advance or regress in the board. Even though they do not add to the inspection itself, they have the purpose of adding to the entertainment, as well as stimulating healthy competition between players.

In order to prevent players from getting too lucky on the dice and skipping too many elements of the model, the following rule was devised: "whenever a player gets the highest number of the dice, she must draw a challenge card". These challenge cards contain instructions to make an additional dice roll, which decides whether the player will (or will not) actually move the full number. Besides increasing the coverage of the session, these cards are used to remind the players about some general situations that, if not considered, may *challenge* a software development project. An example of such a card is "Were the requirements doubts clarified with the stakeholders? Roll the 8-sided dice: if you get less than 5 you can only move half of what you've got before". Only then the player will draw a regular Inspection Card.

Please note that instead of "move 3 spaces", the card reads "move half...". This is the case because the most suitable dice to be used may vary according to the size of the model and to the number of players. For instance, with a small model and only two players, it may be preferable to use a 4-sided dice instead of the usual 6-sided one. In general, a smaller dice will increase coverage but also increase time.

In order to devise suitable heuristics, which would then be translated into inspection cards, the authors could not rely solely on their own expertise. Thus, a literature review was performed in order to identify good modeling practices. Additionally, interviews and protocol analysis with two professors who have authored several i^* papers were performed. This protocol analysis consisted in providing some i^* models to the professors, asking them to inspect the models while explaining their reasoning. Their analyses were video-recorded and then analyzed by us in order to identify their heuristics and then create inspection cards that would enable the players to make similar analyses.

Regarding board game mechanics, some ideas were borrowed from Monopoly, Pictureka and The Game of Life. For instance, when playing a game there seems to be some difference between just "You won $10.000" and "You won a poetry contest! Collect $10.000!" When creating the Inspection Cards, attention was paid to this kind of subtlety in order to keep the game playful, while taking care not to create silly cards since they are expected to be used in a serious context. The full set of inspection cards is available at http://www.cin.ufpe.br/~ler/istargame.

In the next sections, empirical evaluations of the Ring-i process are described. These evaluations aimed to assess the usefulness, enjoyment, ease of use, and acceptance of the process.

4 EVALUATION PILOT

Before green-lighting this gamification approach, two small evaluation sessions were performed, with two volunteers each. Two authors participated in each session: one conducting the evaluation and the other one as an observer.

For this early evaluation, the authors provided the subjects with all the required resources, including the i^* model to be inspected, which was based on a model from the literature [2]. The authors organized the setup, presented the objective of the research, and explained what would happen during the evaluation session. Afterwards, they explained to the subjects the tasks to be performed: the participants had to read the rulebook, read a description of the model to be inspected, and then execute the inspection. Besides observing *in loco*, these sessions were recorded in video for after-the-fact analysis. Moreover, the authors present were not allowed to help with the execution of the inspection, unless when noted that the question raised was not covered in the rulebook – in which case the rulebook would be revised.

After the session, the participants were invited to discuss the positive and negative points of the Ring-i process and to answer a brief feedback questionnaire. All the collected data was analyzed to identify improvement opportunities.

During these early evaluation sessions, the average time to complete the inspection of the i^* model was approximately one hour and twenty minutes. Considering the observed interactions and the result of the feedback questionnaires we could conclude that the participants had fun during the matches, indicating that the game is playful. We identified that the game stimulates the participants to analyze the model seriously while still maintaining the fun factor due to the creativity cards and to other game mechanics. The subjects agreed that the inspection brought improvements for the model – this could be observed during the evaluation sessions and confirmed afterward with the analysis of notes and videos. The participants agreed that the game may be useful and that they would recommend it to other people working with i^* models. A key positive point mentioned by the participants was that the game promotes the analysis and understanding of models without the boredom of traditional analysis methods. A negative point was regarding the coverage of the approach – due to the use of dice rolling, several elements may be skipped during a match. To mitigate this problem, we propose the use of dice with the number of sides appropriate to the model size as well as to the number of participants. Another change aimed to prevent the skipping of elements was the addition of challenge cards, which potentially reduces the amount of movement.

Based on this early feedback, the Ring-i process was improved and a structured empirical evaluation, described next, was conducted.

5 EMPIRICAL EVALUATION

This quasi-experiment is an exploratory qualitative study, aimed at identifying early indications on the applicability of the process and on the quality of the resulting artifacts. It can be characterized as a multi-test within object study, as a single object is examined across different subjects [20]. Its scope can be described as follows:

Analyze the Ring-i process; **for the purpose of** evaluation and improvement; **with respect to** its use by non-experts on i^*; **from the point of view of** software engineers; **in the context of** students applying the process on their own projects.

This quasi-experiment took place on a requirements engineering course within a computer science program, with a sample of 18 undergraduate students. Thus, its context can be classified as a specific context (results cannot be generalized), in an offline environment (i.e., it is not a part of industrial software development), with students working on a toy problem.

In order to ensure that all participants are familiar with i^* models, they had both theoretical and practical classes on i^* modeling. They also had to create i^* models for a software application of their choice, which contributed to their grading. Their participation in the empirical session was completely voluntary as part of their regular classes, no participation-based or performance-based bonus was offered.

5.1 Subjects, Treatments, and Instrumentation

The sample of this study is a set of students from a Requirements Engineering (RE) course offered at the Center of Informatics in the Federal University of Pernambuco, during the first semester of 2017. Thus, this sampling can be classified as a non-probabilistic, convenience sampling [20].

During this RE course, the students had 4 hours of in-person i^* training. As part of the course grading, students were required to create a requirements document for a capstone project, in groups of three to four person. The students voluntarily formed two 3-person groups and three 4-person groups. All students were subject to the same treatment: to enact the Ring-i process to their projects.

The requirements document created by the students necessarily contained an i^* model depicting stakeholders, their relationships, and their intentional elements. After the projects were concluded, these i^* models were made into Ring-i boards by one of the researchers, by numbering the elements and by indicating its start and finish points.

For the evaluation session, each group was given the board depicting their own project, along with a rulebook, and game pieces (tokens and dice). The duration of the evaluation session was 2 hours.

At the end of the evaluation session, their annotations were gathered for analysis and the students were handed a questionnaire. This questionnaire contained six characterization questions, as well as a set of fifteen Likert items aiming to assess usefulness, enjoyment, ease of use, and acceptance of the Ring-i process. In order to encourage honest and unbiased answers, these questionnaires were kept anonymous. Lastly, each group was interviewed in order to obtain qualitative feedback.

5.2 Data analysis

The collected data were processed and organized with a spreadsheet tool. Statistical values and charts were generated with a statistical computing environment (R[1]). Answers to the Likert items of the questionnaire were analyzed through frequency distribution and their median values, which are more appropriate than their average and standard deviation values since they are ordinal data. Consistency between answers was measured by their Cronbach's alpha values [18], which range from zero to one. In general, higher alpha values indicate higher consistency, but larger samples require higher alpha values for the answers to be considered consistent. Lastly, since this is a single-treatment study, no statistical test was performed.

5.3 Threats to Validity

In order to reduce the odds of bias from researchers' expectancies or from hypothesis guessing, the following measures were taken: balanced scales were adopted for the feedback questionnaire; the questionnaire instrument contained multiple questions for the same concept, often reverse-worded, reducing unwanted influence in the answers.

[1]https://www.r-project.org/

Aiming to mitigate researchers' conclusions and publication bias, we opted to present a full breakdown of the answers here discussed, instead of just reporting descriptive statistics. Furthermore, negative results are also included in this paper.

The lack of a control group reduces the value of this empirical evaluation since there is no basis upon which to contrast the results of the treatment group. Moreover, the subjects may have suffered from some fatigue effect, since the empirical session lasted two hours.

Lastly, the small sample size (18 subjects, from 5 groups) damages the external validity of this evaluation.

6 RESULTS

All subjects of this quasi-experiment are undergraduate students enrolled on a Requirements Engineering course. 38.88% of them are Computer Science students, 38.88% are Computer Engineering students, and 22.22% are Information Systems students. Thirteen respondents (72.22%) claimed to have no professional experience with software engineering. The remaining respondents declared the following experience: 9 months, 1 year (two subjects), 2 years, 7 years.

Regarding their gaming experience, 50% of the respondents declared to play board games or cards games rarely or very rarely, 33.33% play occasionally, and 16.67% play frequently. Videogames are played more often, with only 16.67% of the respondents playing rarely or very rarely, whereas 61.11% play frequently or very frequently. Four respondents play videogames occasionally.

All of the systems created and documented by the students are Information Systems. Regarding the size of the i^* models inspected by the subjects, the models had an average of 5.2 actors, with a standard deviation of 1.8. Additionally, they presented an average of 17 dependencies, with a standard deviation of 3.08.

6.1 Usefulness

One way to assess the usefulness of the Ring-i process is by counting the number of changes (improvements or fixes) made to the models as a direct consequence of applying the process. For the five models evaluated in this quasi-experiment, a total of 31 changes were made. This number amounts to an average of 6.2 changes per model, with a standard deviation of 4.14. This indicates that the Ring-i process was useful for finding errors and identifying improvements. The boxplot in Fig. 4 presents the distribution of the number of changes, revealing that a particular model had significantly more changes than the other models (13 changes).

Additionally, the feedback questionnaires answered by each subject provides information on the perceived usefulness of the process: all of the subjects agreed (partially or totally) to the claim that, using the process, they were able to discover errors in the i^* model (Table 1). Furthermore, fourteen respondents (77.77%) disagreed with the reverse worded version of that statement (Table 1).

Surprisingly, even though every subject agreed to have discovered errors, three respondents also agreed that they have not discovered errors in the i^* model. This inconsistency is confirmed by the low Cronbach's alpha value for this pair of questions: 0.40. Analyzing the text of this second statement, we suppose that some respondents may have misread it as "did not find errors on the Ring-i process" rather than the correct "did not find errors *on* the model *with* the Ring-i process". Since the questionnaire was anonymous, we were unable to further investigate the cause of this inconsistency.

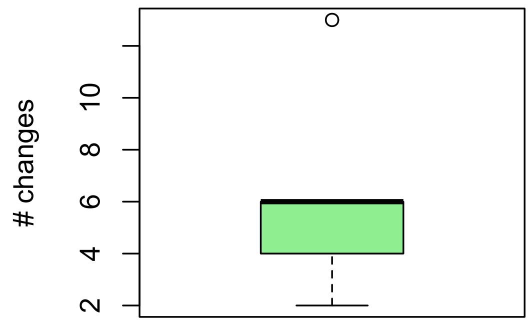

Figure 4: Boxplot representing the distribution of changes resulting from the Ring-i process

Table 1: Questionnaire answers regarding error discovery

	I discovered errors	I haven't discovered errors
1 - Totally Disagree	0 (0%)	10 (55.56%)
2 - Disagree	0 (0%)	4 (22.22%)
3 - Indifferent	0 (0%)	1 (5.56%)
4 - Agree	9 (50.00%)	3 (16.67%)
5 - Totally Agree	9 (50.00%)	0 (0%)
Median	4.5 (between Agree and Totally Agree)	1 (Totally Disagree)

An additional pair of questions asked about improvements to the model, with the following statements: "I was able to identify ways to improve the model with the process"; "The process contributed to the improvement of my model." The results are shown in Table 2. For both statements, the same single respondent was indifferent, whereas all other respondents agreed (partially or totally) with them. The answers to this pair of statements can be considered consistent, as evidenced by their Cronbach's alpha value of 0.75.

Besides considering changes to the model, another dimension of usefulness was evaluated: did the subjects learn something about i^* through the execution of the process? Sixteen respondents (88.88%) agreed (partially or totally) on having learned, whereas one respondent was indifferent and another one did not answer this question (Table 3).

Lastly, the questionnaire contained a Likert item directly probing respondents for the perceived usefulness of the proposal. The respondents unanimously agreed to this statement, as shown in Table 4.

Table 2: Questionnaire answers regarding improvements to the model

	I identified improvements	It contributed to improving the model
1 - Totally Disagree	0 (0%)	0 (0%)
2 - Disagree	0 (0%)	0 (0%)
3 - Indifferent	1 (5.56%)	1 (5.56%)
4 - Agree	9 (50.00%)	10 (55.56%)
5 - Totally Agree	8 (44.44%)	7 (38.89%)
Median	4 (Agree)	4 (Agree)

Table 3: Questionnaire answers on whether respondents learned more about i^* with the process

	I learned something about i^*
1 - Totally Disagree	0 (0%)
2 - Disagree	0 (0%)
3 - Indifferent	1 (5.56%)
4 - Agree	8 (44.44%)
5 - Totally Agree	8 (44.44%)
No answer	1 (5.56%)
Median	4 (Agree)

Table 4: Questionnaire answers regarding usefulness, directly

	It is useful
1 - Totally Disagree	0 (0%)
2 - Disagree	0 (0%)
3 - Indifferent	0 (0%)
4 - Agree	9 (50.00%)
5 - Totally Agree	9 (50.00%)
Median	4.5 (between Agree and Totally Agree)

6.2 Enjoyment

The enjoyment of using the process was assessed through two Likert items in the feedback questionnaire. The first item states that the subject had fun during the process, whereas the second one states that the subject had not had fun at any moment during the process.

The answers to these items are displayed in Table 5. Most respondents agree, partially or totally, that they had fun during the process (83.33%). None of the respondents disagree with this statement. Inversely, most of the respondents disagree, partially or totally, that they had not had fun during the process (83.33%). Two respondents (11.11%) agree with this second statement. Based on the post-experiment interview with each group, some subjects mentioned that repetitiveness was a factor that reduced enjoyment (for instance, one subject complained for having drawn too many challenge cards, which are indeed very similar).

Table 5: Questionnaire answers regarding process enjoyment

	Had fun during the process	Haven't had fun during the process
1 - Totally Disagree	0 (0%)	7 (38.89%)
2 - Disagree	0 (0%)	8 (44.44%)
3 - Indifferent	3 (16.67%)	1 (5.56%)
4 - Agree	8 (44.44%)	2 (11.11%)
5 - Totally Agree	7 (38.89%)	0 (0%)
Median	4 (Agree)	2 (Disagree)

The Cronbach's alpha value for these items is 0.46. Hence, even though the answers for both items paint a similar picture, it cannot be said that they are consistent. Since our sample is small, slight deviations (for instance, some respondents totally agreeing with the first statement but only partially disagreeing with the second one) are enough to reduce this measurement of consistency.

6.3 Ease of use

The ease of using the process was measured by two pairs of Likert items. The first pair directly asked if the process is easy or, inversely, if the process is difficult. The results (Table 6) show that most respondents agreed, partially or totally, that the process is easy (77.78%). No respondent disagreed that the process is easy. Furthermore, 94.44% of respondents disagreed, partially or totally, that the process is difficult. More specifically, no respondent agreed with the statement that the process is difficult. These answers can be considered consistent, with a Cronbach's alpha value of 0.89.

Table 6: Questionnaire answers regarding ease of use, directly

	The process is easy	The process is difficult
1 - Totally Disagree	0 (0%)	6 (33.33%)
2 - Disagree	0 (0%)	11 (61.11%)
3 - Indifferent	4 (22.22%)	1 (5.56%)
4 - Agree	9 (50.00%)	0 (0%)
5 - Totally Agree	5 (27.78%)	0 (0%)
Median	4 (Agree)	2 (Disagree)

The second pair of Likert items aimed to identify whether the respondents understood the proposed process, as well as whether they felt confident about being able to explain the proposed process to somebody else even without access to supporting material. The results are shown in Table 7. All respondents agreed, partially or totally, on having understood the process. However, only 77.78% felt confident on being able to explain it without supporting material. These answers are reasonably consistent, with a Cronbach's alpha value of 0.68.

Table 7: Questionnaire answers regarding ease of use, indirectly

	I understood how to use the process	I could explain the process
1 - Totally Disagree	0 (0%)	1 (5.56%)
2 - Disagree	0 (0%)	1 (5.56%)
3 - Indifferent	0 (0%)	2 (11.11%)
4 - Agree	10 (55.56%)	9 (50.00%)
5 - Totally Agree	8 (44.44%)	5 (27.78%)
Median	4 (Agree)	4 (Agree)

6.4 Acceptance

Three Likert items were used to assess the acceptance of the proposed process, with the following statements: "I wish I learned the proposed process when learning i^*", "The next time I create an i^* model I will use the proposed process", and "I will recommend the proposed process if I meet somebody that is creating i^* models". The answers to these items are displayed in Table 8.

Table 8: Questionnaire answers regarding acceptance of the proposal

	I wish I learned it before	I will use it next time	I will recommend it
1 - Totally Disagree	0 (0%)	0 (0%)	0 (0%)
2 - Disagree	0 (0%)	1 (5.56%)	1 (5.56%)
3 - Indifferent	2 (11.11%)	10 (55.56%)	4 (22.22%)
4 - Agree	11 (61.11%)	6 (33.33%)	10 (55.56%)
5 - Totally Agree	5 (27.78%)	1 (5.56%)	3 (16.67%)
Median	4 (Agree)	3 (Indiff.)	4 (Agree)

Most respondents agreed, partially or totally, with wishing to have learned the process earlier (88.89%), whereas two respondents were indifferent. No respondents disagreed with that statement.

In spite of their positive responses elsewhere in the questionnaire, most respondents were indifferent to the statement regarding using the proposed process again in a next opportunity (55.56%), and one respondent disagreed partially with it. This may be the case because, since they were now familiar with some inspection heuristics and they reportedly learned more about i^* (Table 3), they did not feel the need to use the Ring-i process again. Other possible reason is related to time duration, since the empirical session lasted two hours and the reviewed models were relatively small. Nonetheless, six respondents agreed partially with that statement, and one respondent agreed totally.

Lastly, most respondents agreed, partially or totally, with recommending the process (72.23%). Four respondents were indifferent, and one respondent disagreed partially.

The answers to these three items regarding acceptance of the proposed process are reasonably consistent, with a Cronbach's alpha value of 0.64.

6.5 Discussion

Overall, the results of this empirical evaluation are quite positive, although not definitive. The subjects were able to correctly execute the process, without prior training, and were able to identify several issues in their own models. Moreover, the subjects reported positively for usefulness, enjoyment and ease of use.

Some inconsistencies in the questionnaires' answers, measured with Cronbach's alpha values, were observed. Even though these inconsistencies suggest the need for further evaluation, they are to be expected in such small samples, since slight deviations are enough to reduce this measurement of consistency (for instance, when some respondents totally agree with a statement but only partially disagree with its reverse-worded counterpart).

The results for acceptance were also positive – only one subject (the same) disagreed with its positive statements (Table 8). However, the high percentage of indifference regarding repeated adoption (55.56%) was a surprise to us. This is a key result – even though our usage of gamification provided enjoyment for participants, it did not provide as much impact as wanted regarding repeated adoption. We have identified two possible causes for this: no need of the process once they have learned it, and the time required to apply the process. Because of this result, we plan to develop a non-gamified version of the Ring-i process for evaluation.

On the topic of enjoyment, it has become clear for us that the proposed process achieved its goal of providing enjoyment for its user. This was evidenced not only in the questionnaires' answers, but also by the laughter and friendly teasing that was observed during the evaluation session. Nonetheless, some points of improvement were observed during the post-experiment interview. For some participants, the text of the challenge cards was considered too repetitive, and the cards themselves were considered too punishing.

Lastly, some subjects commented that some cards did not fit their context. This may happen, for instance, with the following card: "The Tasks that are means for this Goal are alternatives, or some may be performed together? If it is the case, the model must be adjusted." If that particular goal happens to be a *dependum*, then it does not have any means-end link, hence the suggested analysis cannot be performed. Another example is the following card "Is this actor a software? If yes, who are its users?" – it the current actor is not a software, then no analysis is performed. The only way we envisioned to handle this drawback is by developing a digital version of the process, which will select suitable cards based on metadata of the model.

7 RELATED WORK

Several research projects have proposed requirements elicitation approaches enriched with gamification techniques. For example, iThink [3] is a "collaborative game-based tool that aims to improve user engagement in the requirements elicitation process." Moreover, REfine [17] is an interactive online platform for elicitation and refinement of requirements through the concept of crowdsourcing.

REfine allows the distribution of tasks to a large number of unknown users on a large scale through public calls for contribution [15]. In [19], the focus is on eliciting and validating requirements with a gamified application that supports such activities. Features, such as online chat rooms, enable participants to discuss issues related to their requirements. In addition, stakeholders can post comments on requirements written by other stakeholders to collaborate on writing a high-quality document, encompassing their possibly different views. Whereas these approaches are online and general, our proposal is offline and tailored specifically to i^* models inspection.

More recent works have tried to use modeling concepts with the objective of creating platforms, languages, and frameworks that aid in the gamification process. The GaML project [7], for example, elaborated a formal domain-specific language to define gamification concepts. The primary design goal for the GaML is for it to be readable by domain experts such as consultants or designers with minor IT background. Similarly, we expect the Ring-i process to be understandable to untrained users, even though the presence of at least one i^* expert is advised.

The game for ArchiMate described in [5] has some resemblance to our proposal, in the sense that both use a model as the game board. Nevertheless, they differ in the sense that the ArchiMate game (i) requires a significant effort by a game master to prepare the game for a specific model and (ii) is targeted to architectural models. In contrast, the Ring-i process (i) requires minimal preparation and (ii) is targeted to requirements models (i^*).

The topic of i^* models verification and validation is understudied. The VeMI approach [4] verifies i^* models in the context of Model-Driven Development (MDD). There, the model is verified indirectly by analyzing a target model generated from a source i^* model. Here, the verification is not tied to any specific development approach.

8 CONCLUSIONS AND FUTURE WORK

This paper presents Ring-i, a gamified process for inspecting i^* models. An empirical evaluation of the proposed process provides evidence that the process is perceived as useful, enjoyable, and easy to use. Nonetheless, acceptance of the proposal is unclear, since most of the subjects do not agree with the statement that they would use it the next time they create an i^* model. Based on this result, we plan to create and evaluate a non-gamified version of the Ring-i process.

In orded to evolve the Ring-i process itself, we expect to perform additional empirical evaluation and gather further feedback. Additionally, we intend to provide tool support for the Model Preparation task, on which the i^* models are edited in order to resemble a game board.

Another venue for future work is the customization of the Ring-i process in order to support the inspection of other requirements representations, such as Business Process Model and Notation, Use Cases, textual requirements, as well as other i^* variants, such as Tropos and i^* 2.0.

ACKNOWLEDGMENTS

The authors would like to thank all the subjects that partook in the empirical evaluation here reported. This project was partially funded by the following funding agencies: CNPq and FACEPE.

REFERENCES

[1] Thomas E Bell and Thomas A Thayer. 1976. Software requirements: Are they really a problem? *Proceedings of the 2nd international conference on Software engineering* (1976), 61–68.
[2] Jaelson Castro, Manuel Kolp, and J Mylopoulos. 2002. Towards Requirements-Driven Information Systems Engineering: The Tropos Project. *Information Systems* 27, 6 (2002), 365–389.
[3] João Fernandes, Diogo Duarte, Claudia Ribeiro, Carla Farinha, João Madeiras Pereira, and Miguel Mira da Silva. 2012. iThink: A game-based approach towards improving collaboration and participation in requirement elicitation. *Procedia Computer Science* 15 (2012), 66–77.
[4] Giovanni Giachetti, Beatriz Marín, Lidia López, Xavier Franch, and Oscar Pastor. 2017. Verifying goal-oriented specifications used in model-driven development processes. *Information Systems* 64 (2017), 41–62.
[5] Jos Groenewegen, Stijn Hoppenbrouwers, and Erik Proper. 2010. Playing archimate models. *Enterprise, Business-Process and Information Systems Modeling* (2010), 182–194.
[6] Juho Hamari, Jonna Koivisto, and Harri Sarsa. 2014. Does gamification work? – a literature review of empirical studies on gamification. In *47th Hawaii International Conference on System Sciences (HICSS)*. IEEE, 3025–3034.
[7] Philipp Herzig, Kay Jugel, Christof Momm, Michael Ameling, and Alexander Schill. 2013. GaML-A modeling language for gamification. In *Utility and Cloud Computing (UCC), 2013 IEEE/ACM 6th International Conference on*. IEEE, 494–499.
[8] Jennifer Horkoff, Tong Li, Feng-Lin Li, Mattia Salnitri, Evellin Cardoso, Paolo Giorgini, John Mylopoulos, and Joao Pimentel. 2014. Taking goal models downstream: a systematic roadmap. In *Research Challenges in Information Science (RCIS), 2014 IEEE Eighth International Conference on*. IEEE, 1–12.
[9] Mohamad Kassab. 2015. The changing landscape of requirements engineering practices over the past decade. *2015 IEEE Fifth International Workshop on Empirical Requirements Engineering (EmpiRE)* August 2015 (2015), 1–8. https://doi.org/10.1109/EmpiRE.2015.7431299
[10] Martina Z Huber Kolpondinos and Martin Glinz. 2017. Behind Points and Levels - The Influence of Gamification Algorithms on Requirements Prioritization. *Proceedings of the 25th IEEE International Requirements Engineering Conference* (2017). https://doi.org/10.1109/RE.2017.59
[11] Philipp Lombriser, Fabiano Dalpiaz, Garm Lucassen, and Sjaak Brinkkemper. 2016. Gamified Requirements Engineering: Model and Experimentation. In *Proceedings of the 22nd International Working Conference on Requirements Engineering: Foundation for Software Quality. LNCS*. Vol. 9619. 171–187. https://doi.org/10.1007/978-3-319-30282-9_12
[12] Jane McGonigal. 2011. *Reality is broken: Why games make us better and how they can change the world.* Penguin.
[13] Oscar Pedreira, Félix García, Nieves Brisaboa, and Mario Piattini. 2015. Gamification in software engineering–A systematic mapping. *Information and Software Technology* 57 (2015), 157–168.
[14] Luca Piras, Paolo Giorgini, and John Mylopoulos. 2016. Acceptance Requirements and Their Gamification Solutions. *Proceedings - 2016 IEEE 24th International Requirements Engineering Conference, RE 2016* (2016), 365–370. https://doi.org/10.1109/RE.2016.43
[15] Asarnusch Rashid, David Meder, Jan Wiesenberger, and Astrid Behm. 2006. Visual requirement specification in end-user participation. In *Multimedia Requirements Engineering, 2006. MERE'06. First International Workshop on*. IEEE, 6–6.
[16] Sachidanandam Sakthivel. 1991. A survey of requirements verification techniques. *Journal of Information Technology* 6, 2 (1991), 68.
[17] Remco Snijders, Fabiano Dalpiaz, Sjaak Brinkkemper, Mahmood Hosseini, Raian Ali, and Atilla Ozum. 2015. REfine: A gamified platform for participatory requirements engineering. In *Crowd-Based Requirements Engineering (CrowdRE), 2015 IEEE 1st International Workshop on*. IEEE, 1–6.
[18] Mohsen Tavakol and Reg Dennick. 2011. Making sense of Cronbach's alpha. *International journal of medical education* 2 (2011), 53.
[19] Naomi Unkelos-Shpigel and Irit Hadar. 2015. Inviting everyone to play: Gamifying collaborative requirements engineering. In *Empirical Requirements Engineering (EmpiRE), 2015 IEEE Fifth International Workshop on*. IEEE, 13–16.
[20] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, and Anders Wesslén. 2012. *Experimentation in software engineering*. Springer.
[21] Eric Yu, Paolo Giorgini, Neil Maiden, and John Mylopoulos. 2011. *Social modeling for requirements engineering*. Mit Press.