Taking Seriously Software Projects Inception Through Games

Miguel Ehécatl Morales-Trujillo[1](✉), Hanna Oktaba[1], and Juan Carlos González[2]

[1] KUALI-KAANS Research Group,
National Autonomous University of Mexico, Mexico City, Mexico
{migmor,hanna.oktaba}@ciencias.unam.mx
[2] ENTIA, Av. López Mateos 2077-Z16, Col. Jardines de Plaza del Sol,
Guadalajara, Mexico
jgonzalez@entia.com.mx

Abstract. Inherent properties of games, such as rules, goals and interaction, have made them popular to address challenges and sort obstacles in a wide variety of contexts. Within Software Engineering, a challenging activity of the software development process is the Inception phase, in which stakeholders' needs, required functionalities, objectives, risks and constraints of a software product are established. An alternative to optimize stakeholders' participation during the Inception phase and make more efficient its outcomes is to do it through games.

Taking into account the uncommonness of games in Software Engineering development process and the lack of complex methods that include games, this paper presents ActiveAction, a method that combines classical requirements specification techniques with games. ActiveAction resulted in a successful game-based strategy that has improved the Inception phase of the software development projects based on the fact that stakeholders express their ideas freely in an unstressed environment; real-life scenarios are simulated to identify exceptions; requirements and risks are determined in a collaborative manner. It is concluded that the inclusion of games in such a challenging activity as software projects inception, is feasible and reported promising results that benefit both stakeholders and software developer organizations.

Keywords: Games · Inception phase · Requirements specification · Stakeholders' involvement · Software project · Workshop

1 Introduction

A game could be defined informally as an amusement activity with rules, undertaken for entertainment but still pursuing a goal. A game can be seen as an activity in which a player must learn new skills and apply them to overcome challenges, getting rewards or punishments, depending on their success or failure, respectively. The application of games in education, government or military areas has become an important and feasible alternative to face challenges. Since the entertainment is not the main purpose of these games, they are labeled as serious games.

© Springer International Publishing Switzerland 2015
J. Filipe and L. Maciaszek (Eds.): ENASE 2014, CCIS 551, pp. 109–124, 2015.
DOI: 10.1007/978-3-319-27218-4_8

A serious game is a game that does not have entertainment, enjoyment or fun as their primary purpose [1]. Even thought, according to [2] games may be played seriously or casually, it does not mean that serious games are not, or should not be, entertaining.

Nowadays, serious games are becoming more popular and have a broad scope of usage solving business problems and challenges that face the public sector [3]. The ludic and playful aspects inherent to games, serious or not, can be used as a strategy to address different kind of challenges and sorting obstacles. Enclosing to Software Engineering, it can be asserted that developing software and systems is a challenging activity. In 2010, Standish Group, through CHAOS Report [4], revealed that only 32 % of software projects were successful, 44 % challenged and 24 % were cancelled. Comparing these data with that of other industries, Construction for example, 94 percent of the customers were satisfied with the results of their projects, which suggests that construction projects have much lower failure rates than software projects [5].

But, why is developing software different to producing any other product? Why is it so difficult to develop good products in a consistent way from their inception phase and on? The purpose of the Inception phase of a software projects is establish the business case for the system and delimit the project scope [6]. The stakeholders' involvement in this phase is inherent and fundamental.

This necessity for stakeholders' involvement becomes evident if we analyze the CHAOS reports. Since 1994 the top three main success factors in projects are:

- Executive Support.
- User Involvement.
- Clear Business Objectives.

The three of them are evidently related to the stakeholders and the Inception phase of the project.

However, including stakeholders as part of the team, but in the real sense, requires time and money. With that in mind this paper presents ActiveAction, a workshop used as an alternative into the software project's Inception phase. ActiveAction combines classical with game-based techniques to improve stakeholders' involvement in the Inception phase of a project and increase its effectiveness.

The paper is organized as follows: Sect. 2 presents main success factors in IT projects, while Sect. 3 expands on the strategies to conceptualize projects using traditional, agile and game-based techniques. Section 4 then goes on to introduce and analyze the ActiveAction workshop, while Sect. 5 presents results of the usage of this method, its advantages and disadvantages. Conclusions and future work are provided in Sect. 6.

2 Success Factors in Software Projects

Are software projects different to other projects? According to [5] the main difference relies on the characteristics of software and technology. On one hand, software is abstract and complex while the requirements to be developed are incomplete. On the other hand, the vast domain of technology changes rapidly, causing lack of technical experience and immaturity of best practices.

According to Gartner [7] the most common “anti-success” project factors are: poor quality, cancelling after launch, high cost variance, substantially late delivery and functionality issues. Determining if a project is successful or not could be hard, the success can be determined by a combination of factors like scope, time or cost, but another one could be ROI, quality or the healthiness of the workplace [8].

However, failure aspects related to technical issues are not a critical factor [9]. It means that improving technical aspects of projects will not always lead to higher ratios of success. Moreover, studies like [10] found that:

- Business is usually or always out of sync with project requirements.
- Stakeholders need to be more involved and engaged in the requirements process.
- Business objectives are fuzzy.
- Requirements definition processes do not reflect business needs.

As it can be noticed, the “theory” of what to look for is clear, the problem is how to achieve it.

Building robust business-oriented requirements can be half of the solution; according to [11] one group of success factor is to manage strategy and stakeholders. In other words, the focus should be on obtaining clear business objectives, well-defined business cases, major stakeholders’ alignment, a stable scope and executive support instead of concentrating on budget and scheduling. Taking these ideas into account, a first approach is to focus on the very beginning of the project, and increase the stakeholders’ involvement in order to improve business modeling and specification of requirements.

Nevertheless, the stakeholders’ involvement implies investing time, money, experience and patience from both parties: software developer organization and stakeholders.

Furthermore, one of the major problems with most business engineering efforts, is that the Software Engineering and the Business Engineering community do not communicate properly with each other [6]. If this is true, increasing the stakeholders’ participation won’t be a solution. But, what happened if both communities started to talk in another language? For example using games as a way of communication.

We believe that inclusion of games is a beneficial strategy to conceptualize a project together with the stakeholders and increase the success rate of projects.

3 Software Projects Inception Phase

Conceptualization begins at the early steps of projects, in the words of RUP [6], it occurs during the Inception phase. At the Inception phase the business process is documented in order to assure a common understanding among all stakeholders, the system is described determining the required functionalities, objectives, risks and constraints, to deliver, successfully, a product which meets the stakeholders’ needs.

The Inception phase includes two core process workflows: Business Modeling and Requirements. And one core supporting workflow: Project Management.

During the Business Modeling workflow, the business process is documented using use cases in order to assure a common understanding among all stakeholders.

The Requirements workflow is in charge of describing what the system should do and allows the developers and the stakeholders to agree on that description. To achieve this, the required functionalities and constraints should be elicited, organized and documented.

Lastly, Project Management workflow is the art of balancing competing objectives, managing risks, and overcoming constraints to deliver, successfully, a product which meets the stakeholders' needs [6].

The outcomes of the Inception phase, defined by RUP, are:

- A vision document: a general vision of the core project's requirements, key features, and main constraints.
- An initial use-case model.
- An initial project glossary may optionally be partially expressed as a domain model.
- An initial business case, which includes business context, success criteria, and financial forecast.
- An initial risk assessment.
- A project plan showing phases and iterations.
- A business model, if necessary.
- One or several prototypes.

In order to achieve and create these outcomes, during an ordinary Inception phase, development teams deal with plenty of meetings with different stakeholders, final users and management staff getting a broad idea of their sometimes contradictory needs. Business analysts start to mediate inconsistent requests and the work that was intended to take a couple of weeks becomes longer.

A feasible solution, applied during Inception phase, could be to elicit, envision, design and prioritize the business needs until a minimal viable product is clearly seen by key stakeholders involved in the project, collaborating with the development team.

In light of this, three approaches to manage activities during the Inception phase are presented in the next subsections.

3.1 Agile Approach

Small releases and stakeholders', users' and customers' involvement in day-to-day activities are part of the philosophy of agile practices to face the requirements elicitation and prioritizing.

According to agile practices, conceptualization and development of a project occurs almost at the same time. It is achieved thanks to the Product Owner (PO), a role in charge of having a vision of what to build: they create and prioritize a list of desired features, the Product Backlog. PO must have a solid understanding of users, market place, competition, and future trends for the domain or type of system being developed. Also the PO requires working closely with key stakeholders [12].

An important complication of agile approach is that the PO could be more than one person, and most important, not all of them are ready or available to be involved in a day-to-day dynamic.

3.2 Traditional Approach

The traditional approach is a process-centered way of working. First, adequate stakeholders are identified. Later, fixed interviews with clients and stakeholders are held in order to identify their needs. The input of this process is obtained from meetings, focus groups, questionnaires or user observations. The information is transformed by the development team until a business model or requirements specification is created and agreed upon with stakeholders.

This approach is expensive and demands an arduous effort from both parties.

3.3 Game-Based Approach

Inception is a suitable phase to include stakeholders as a real part of the team and not only as a source of needs. They can be involved in depth because of the non-technical nature of the phase activities.

Key concepts of games such as goals, rules, challenge and interaction are also present in several real-world activities, for example a structured software development process [13].

On the other hand, developing software is a challenging activity that is seldom regarded as fun. Just like many other types of activities, it can be organized as a set of hierarchical and partially ordered challenges that must be overcome, often requiring several different skills from developers, and lots of teamwork effort [14].

In fields of industry, such as marketing and sales, games are used to help to understand customers, market, business opportunities and to improve everyday employee-related issues within an organization [15]. The application of these games has a broad spectrum: some examples taken from [16] define their variety. For instance, Empathy Map's objective is to gain insight and understanding for a targeted public image; How-Now-Wow Matrix pursuits the goal of selecting the best ideas as a group; Job or Joy aims at improving working environment of the organization; lastly, there are classical techniques, such as SWOT analysis and Pros/Cons adapted as games.

On the other hand, the application of such games tends to be isolated strategies in order to accomplish specific goals that may not be related to Software Engineering. Besides, they are not organized as a interrelated set of activities that reuses the result of a previous activity as an input for the next, thus making difficult to compose a method to attack more complex problems.

Few reports of games that support approaches to perform Software Engineering activities with objectives not related to education nor training can be found. One of those games is Software Quantum Metaphor, a game that makes an allusion to the chemical units known as quantum. The requirements of the software project are visualized as a bag of balls. Each ball visualizes one requirement or quantum. There are balls of different colors, each color representing a different state of analysis of the requirements [17].

Another game is the Labor Game Method, described in [18]. In this game the requirements are presented through cards on a board. Each card can be changed, or some cards can be added or removed from the board. This game permits to visualize all the requirements of the project, their structure and priorities.

Taking into account the uncommonness of games in Software Engineering development process and the lack of a method that puts together games in order to accomplish complex goals, the next section presents a method that combines a set of classical techniques with games, and is intended to improve the Inception phase of software projects.

4 ActiveAction Workshop

Entia [19] is a Mexican IT organization established in 2003 as a software developer organization conformed at the beginning by 4 employees, now 20. Entia has executed projects to construct custom software related to finance, retail, manufacturing and service sectors. They can be classified into "process systematization" and "business venture". The former consists in developing a software system that automates a specific process in the stakeholder's organization. The latter is aiming at adding a new service or product that will be offered to the stakeholder's clients.

In order to increase the projects' success rate and, consequently, the number of projects, Entia developed Innocamp strategy. Innocamp consisted in an intense workshop where all stakeholders collaboratively design their new custom software development project. This envisioning strategy was created in order to avoid continual coming and going between customers and Entia work team, caused by the lack of definition, clarity and consensus in their needs.

The Innocamp workshop let the organization achieve such benefits as: bringing all of the stakeholders to the organization at the same time and place; stakeholders and work team invested quality time during the workshop; complexity and size of a project was better understood by the stakeholders; and both parties developed synergy and emotional attachment to the project.

Since 2007 Innocamp has evolved into ActiveAction workshop. The evolution started with the inclusion of games as a way of entertaining the stakeholders during the workshop. Over the time the work team noticed that games helped to decrease resistance and increase concentration level of the stakeholders, causing an important impact on the results of each workshop.

ActiveAction is a game-based workshop focused on the software project conceptualization. It consists of the following activities:

- **Pre-Day:** It enables Entia to figure out stakeholders' real motivations that made the new project feasible. It includes negotiations between the customer and the organization, calendar planning for the rest –Days and collecting relevant customer information. The organization chooses adequate participants to take part in the following –Days.
- **Intensive-Day:** It is the actual method that gathers stakeholders and consultants during 8-12 h in the same place. 5 roles are required for the consultant team: Coach, Analyst, Process Engineer, Logistics and Customer Service and Support. The purpose of the Intensive-Day is to extract expectations, objectives, needs, risks, and functional and non-functional requirements from stakeholders using a collaborative and game-based strategy.

- **Post-Day:** The information gathered during Pre- and Intensive-Day is structured in order to create an IT strategy to deal with the project.
- **Action-Day:** on this –Day a meeting takes place in which the context, technological strategy and supplier proposals are discussed with the stakeholders in order to define the path that a project will take.

For the purposes of this paper, the Intensive-Day is presented in detail in the next subsection.

4.1 Intensive-Day Method

To carry out the Intensive-Day practices the following requirements and material are supplied: a room with 4.5 square meters per participant; speakers, a whiteboard, a glass wall or canvas, a projector, a TV, a special table for personal belongings and toys, such as little balls and office supplies like post-its, tacks, pencils and markers. Additionally each consultant needs a computer, Internet connection, source of electrical power and connection to the projector. All participants are seated in circle.

Intensive-Day method is composed by 19 practices that can be classified in two groups:

- **Core** practices produce meaningful outcomes for the Inception phase of the project and have a defined order.
- **Auxiliary** practices are focused on keeping order, reducing stress and maintaining high entertainment level. They have no special order to follow.

Figure 1 presents the Intensive-Day method where the game-based practices are marked with a “balero” icon, which is a traditional Mexican toy. The core practices are colored in light green, while the auxiliary are dark green. Flows between practices indicate the order of their execution.

The Intensive-Day method was expressed as a composition of practices using KUALI-BEH approach [20]. This approach establishes that a practice provides a systematic and repeatable way of work focused on the achievement of an objective. Each practice pursues the objective of producing a result that originates from an input.

Using these three elements, objective, input and result, the Intensive-Day game-based practices are described below.

4.2 Game-Based Practices

This section presents 10 game-involving practices.

Cover Story and Product Box (Core). The objective of the practice called Cover Story and Product Box [21] is to identify and express the stakeholders’ objectives. An interesting fact is that the identified objectives should be expressed as if they had already been achieved.

The practice of Cover Story is suitable when the stakeholders' objectives belong to the area of improving their companies' image, functioning or prestige. The practice of Product Box is more appropriate if the stakeholders' objective is to own and sell a particular product.

In the former the stakeholders have to design a cover story for a magazine that would be published in the future and would narrate their success story; while in the latter they design a box that would contain the final product.

The input of these practices is objectives provided by the stakeholders, while the results are the cover story or the product box with comprehensive objectives.

Persons (Core). The objective of Persons is to identify priorities of cost, time and scope and their control strategy.

Three persons from the stakeholders' team portray each one of those variables, highlighting the pros and cons in order to clarify the importance of reducing or increasing their values. The stakeholders listen and watch the game and have to make a decision at the end, putting the three variables in a priority order.

The input of the practice is an example of a variable control matrix while the result is a matrix with the agreed values of cost, time and scope.

Role Play (Core). The Role Play objective is to simulate the organization's everyday process. The simulation consists in role-playing performed by stakeholders.

The role-play scenario is based on a frequent, but complex episode from everyday business-oriented activities in the stakeholders' organization. It also should have exceptions or miscommunication issues and needs to be rebuilt from a happy path.

The input of this practice is the actual process of the organization used as a role play script. The results are: a matrix of involved roles and a successful criteria list of an everyday process. It also provides contextual information of the stakeholders' organization.

Metaphor (Core). The Metaphor practice is based on the well-known game Lego Serious Play, developed by LEGO [22]. It is a structured process, where participants use LEGO bricks to create models that express their thoughts, reflections and ideas [23]. Here the prime objective is to obtain product expectations from every member of the stakeholders' organization.

The input is the matrix of involved roles and a metaphor to motivate generation of ideas that will be expressed as a LEGO figure. Each person of the stakeholders team builds a figure, explains its meaning and all of them are put together to interpret their relationships and interactions working as one team.

The results are: an interpretation of each LEGO figure and a better understanding of relationships among the roles in the matrix and within the process through the set of LEGO figures.

Speedboat (Core). The game practice of Speedboat is adapted from the game designed by Luke Hohmann [21] and its objective is to identify the project's potential risks by and for the whole stakeholders' team.

A speed boat is drawn on the top part of a whiteboard while the participants write any risks that can affect the project on post-its. Then they place them on the white board as if they were anchors. The lower a post-it is placed, the heavier the anchor is, which means a high risk that will impact the project the most. After all the "anchors" are set, a moderator groups similar ones together and the participants discuss the identified risk factors. The probability of occurrence is determined by the number of repeated risks and the impact by the lowest position.

The input of this practice is post-its used by each person, while the result is the list of identified risks associated with their probability of occurrence and impact.

AVAX Storming (Core). Based on the popular brainstorming technique [24], the practice of AVAX Storming is focused on finding the desired functional requirements for the system. All the participants write a one functional requirement on a piece of paper named Added Value Actionable Items (AVAX). Later AVAXes are grouped in order to sketch the system future modules.

This practice helps users to figure out the size of their project because soon the walls start to be filled up and the size of the project starts to grow. It is a visual way to show users that each need has its complexity and weight. If the project has a limited budget, each AVAX has a section where each person chooses if the requirement is "Needed" or "Desired". When all AVAX are posted on the wall, each person explains in detail their AVAX to the rest of the team opening it to discussion.

The result of the practice is a mind map of the identified AVAXes.

Buy a Feature (Core). The objective of the practice called Buy a Feature is to determine the scope of what will be built by the end of a single iteration. Buying a feature with a fixed budget is oriented to identify the top-rated requirements. The stakeholders' team has to buy the most necessary requirements and define the Minimum Viable Product (MVP).

The input of the practice is the list of required functionalities or AVAX, each with a price tag and fake bills for each stakeholder.

The result is the MVP or a list of AVAXes bought by the stakeholders' team. The MVP is also helpful to validate the project's feasibility.

Gunfight (Auxiliary). During ActiveAction there are "rules" to be followed, such as not to use smart phones or not to leave the room except during breaks. If anybody disobeys the rule, the rest of the participants can shoot at them with NERF guns until the "order" is restored, which is actually the objective of the practice.

Time Police (Auxiliary). The stakeholders' team chooses a person who will play the Time police role. He or she has a vuvuzela (horn) that is blown when somebody is talking about a closed topic or is digressing. The objective of the practice is to maintain discussions focused and to watch the time.

Circle of Trust (Auxiliary). The practice of Circle of Trust pursuits the objective that anybody in the room is allowed to express their opinions. To start a circle of trust, the interested person rings a bell and puts forward a topic. Every single person in the room has to express their sincere opinion on the question without being pressured.

4.3 Non-ludic Practices

This section presents the rest of the practices carried out during the Intensive-Day. Making a total of 9, they have no ludic aspects included.

- **Preparation:** planning and organizing the logistics required for the Intensive-Day.
- **Opening and Welcoming:** opening the day and presenting schedule, rules and logistics to the stakeholders.
- **Team Roles:** introducing the teams and explaining the roles to be used.
- **Concepts:** explaining concepts, dynamics and supporting tools to be used during the -Day.
- **Enclosing:** stakeholders define initial objectives, individual expectations and expand on the actual reasons for being there.
- **Packages:** categorizing needs and requirements expressed by the stakeholders. Identified packages are used to initially group desired functionalities obtained during AVAX Storming practice.
- **Non-functional Requirements:** determining the non-functional requirements of the desired system. Consultants use a fixed questionnaire based on the Pre-Day information.
- **Final Agreements:** closing all open topics, dispelling doubts and getting the last comments.
- **Closure and Raising a Toast:** applying a satisfaction survey of the -Day and making a farewell toast.

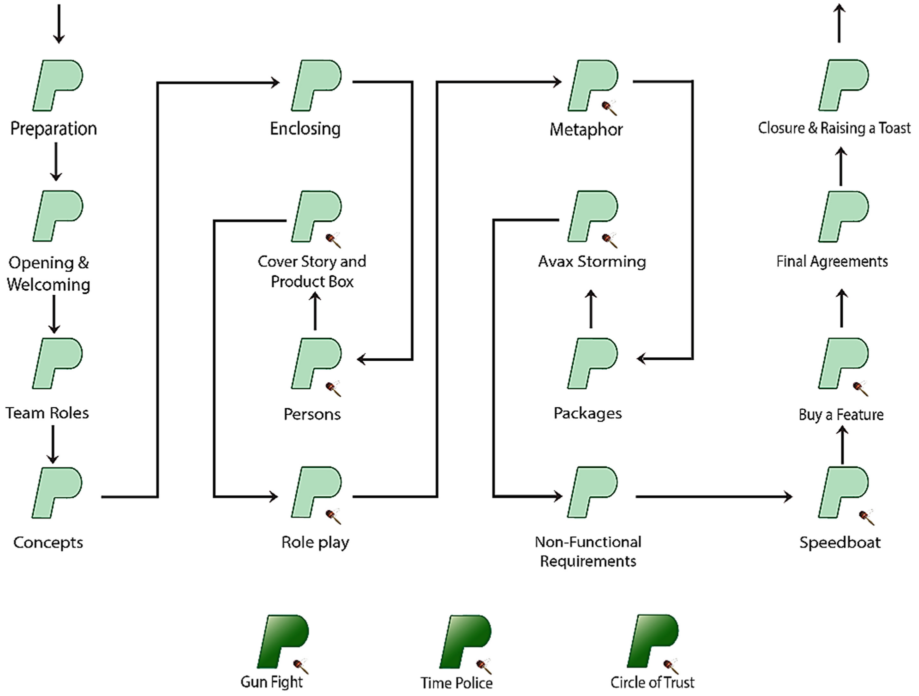

Fig. 1. Intensive-day method diagram.

4.4 Intensive-Day Work Products

During Intensive-Day the Coach guides practices' execution while the Analyst and the Process Engineer are responsible for concentrating the results of each practice in the following work products:

- Contextual information and Everyday Process.
- Stakeholders' expectations.
- Objectives and priorities.
- Functional requirements.
- Strategy and work to be done.
- Non-functional requirements.
- Risks analysis.
- Consequences of undertaking the project for the developing organization.
- Minimum Viable Product.
- Supplier proposals summary.

A summary of the work products is concentrated in a mind map. The left part of the mind map provides a description of each work product, while the right part summarizes the Functional Requirements as AVAXes, see Figs. 2 and 3.

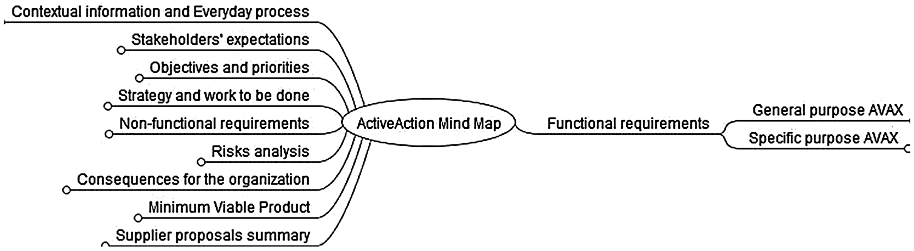
Contextual information and Everyday process
Stakeholders' expectations
Objectives and priorities
Strategy and work to be done
Non-functional requirements
Risks analysis
Consequences for the organization
Minimum Viable Product
Supplier proposals summary
ActiveAction Mind Map
Functional requirements
General purpose AVAX
Specific purpose AVAX

Fig. 2. ActiveAction mind map.

Based on the objective of each practice, the generated work products can be associated to the Inception phase outcomes defined earlier. Table 1 shows the association between outcomes and work products. The Intensive-Day practice that generates the respective work product is shown between parentheses.

In an intent to verify the usefulness of the Intensive-Day products, RUP evaluation criteria were taken into account. According to [6] the following criteria were identified:

- Stakeholder concurrence on scope definition and cost/schedule estimates.
- Requirements understanding as evidenced by the fidelity of the primary use cases.
- Credibility of the cost/schedule estimates, priorities, risks and development process.

The mapping between outcomes shows that the stakeholders agree on the scope definition through *Functional requirements* and *Minimum Viable Product*, also top requirements are prioritized. Besides, an initial cost and schedule estimation can be derived from *Strategy and work to be done*.

Table 1. Mapping of outcomes.

Inception phase outcomes	Intensive-day outcomes
A vision of the project's core requirements, key features, and main constraints	Contextual information (metaphor)
An initial use-case model	Functional requirements (AVAX storming)
An initial project glossary or a domain model	Contextual information (metaphor) Stakeholders' expectations (persons)
An initial business case, which includes business context, success criteria, and financial forecast	Objectives and priorities (cover story or product box) Stakeholders' expectations (Persons)
An initial risk assessment	Risks analysis (speedboat)
A project plan showing phases and iterations	Strategy and work to be done (mind map)
A business model, if necessary	Everyday process (role play)
One or several prototypes	Minimum viable product (buy a feature)

Understanding of requirements is achieved through the AVAXes, which are initially defined by the stakeholders' team. Priorities are obtained using *Objectives and priorities* which are expressed directly by stakeholders. Identified objectives and priorities consequently affect the selection of features of the *Minimum Viable Product*. Later, risks are identified and weighed during *Risk Analysis*. Finally the credibility of the resulting work products predicts to be high due to the fact that they were created with active participation of the stakeholders.

4.5 AVAX Mind Map for Project Tracking

The AVAX mind map, in particular its right part which contains the functional requirements in the form of AVAXes, is used along the project as a tool to track its health and progress.

For this purpose each AVAX has a set of valid states:

- **Identified:** applies to all AVAXes identified during the Intensive-Day.
- **Posterior:** applies to AVAXes that are added by the stakeholder after the Intensive-Day. A "light bulb" icon is used to represent it.
- **Deleted:** applies to an AVAX which is not needed anymore or is outside the priorities of scope, cost or time. A "No trespassing" icon is used to represent it.
- **Delivered:** applies to an AVAX which is passed on to the stakeholders. A "check" icon is used to represent it.
- **Validated:** applies to an AVAX which is approved by the stakeholders and gets a "smiley face" icon.
- **Externally Done:** applies to an AVAX which is provided by an external organization. An "exclamation mark" icon represents this state.

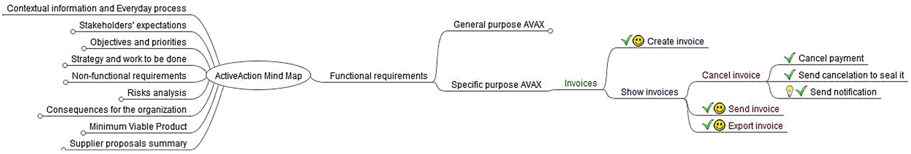
Contextual information and Everyday process
Stakeholders' expectations
Objectives and priorities
Strategy and work to be done
Non-functional requirements
Risks analysis
Consequences for the organization
Minimum Viable Product
Supplier proposals summary
ActiveAction Mind Map
Functional requirements
General purpose AVAX
Specific purpose AVAX
Invoices
Create invoice
Show invoices
Cancel invoice
Send invoice
Export invoice
Cancel payment
Send cancelation to seal it
Send notification

Fig. 3. AVAX detail and status.

Figure 3 shows *Invoices* package (in green) that contains the AVAXes *Create invoice* and *Show Invoices* (in blue) which is decomposed into three branches: *Cancel invoice*, *Send invoice* and *Export invoice* (in red). The branch *Cancel invoice* generates three more branches: *Cancel payment*, *Send cancelation* to seal it and *Send notification* (in black). Their states with their respective icons can be observed in the map.

To maintain trustworthy relationships between stakeholders and development team, the AVAX Mind Map used by the organization for project development and tracking must be exactly the same as the one generated during the Intensive-Day.

5 Results

ActiveAction workshop resulted in a successful game-based strategy that has improved the Inception phase of the software development projects based on the fact that: stakeholders express their ideas freely in an unstressed environment; simulate real-life scenarios to identify exceptions; prioritize risks and identify requirements in a collaborative manner.

ActiveAction has been applied in 19 projects with the following time distribution of the Intensive-Day: 65 % dedicated to game-based core practices, 17 % of resting time and 18 % of non-ludic practices.

5.1 ActiveAction Advantages and Disadvantages

ActiveAction has resulted in a powerful tool for Entia, which started to sell it as a separate service, giving the customer the right to choose Entia or another organization for further construction.

The following are ActiveAction workshop benefits identified by Entia:

- Software products' origin and rationale are known and understood by stakeholders.
- Each and every of the stakeholders' points of view are taken into account.
- The whole team takes the project as something of its own, not something imposed.
- Relatively relaxed in-game environment helps to come to agreements among many people.
- Applying games helps to establish rules, objectives and behaviors in a natural way.
- Customer satisfaction and emotional engagement is easily achieved.
- The Inception phase is completed within three weeks.
- The Entia successful project rate has grown 25 percentage points, from the initial 56 % up to 81 %.

On the other hand, ActiveAction requires a high demand of skills, concentration and knowledge from the Coach, Analyst and Process Engineer roles. 12 h in-a-row, independently of the activity to do, requires a lot of effort and concentration, so finding adequate pace is important. Besides, the organization must distribute carefully their key personnel, consultant team members, between their daily work and the ActiveAction Days, which can compromise other projects of the organization.

The following are drawbacks identified by Entia:

- Difficulty in getting stakeholders together if they are geographically distributed.
- Difficulty involving stakeholders in the –Day's activity due to their passivity.
- Uneasiness to share opinions when the "boss" is present.
- Including stakeholders' clients into the Intensive-Day may not always be possible.

5.2 Validation and Improvement Suggestions

In order to get feedback on limitations and identify opportunities of improvement, three surveys are applied both to stakeholders and consultants. A scale from 1 to 5 is used to rate each item, also a free comment section is also considered.

Two of the surveys are applied at the end of the Intensive-Day with the objective to measure stakeholders and consultants' satisfaction. The third survey is applied to stakeholders at the end of the Action-Day. The grades average from the last 10 ActiveActions is summarized in Table 2.

Table 2. Surveys' results summary.

Intensive-day stakeholders' survey	**Average**
Overall experience	4.8
Achievement of expectations	4.8
Intensive-day consultants' survey	**Average**
Overall experience	4.4
Stakeholders team involvement	4.7
Consultants team performance	4.7
ActiveAction stakeholders' survey	**Average**
Proposed solution	4.6
Has ActiveAction benefited your organization?	4.7
Would you use ActiveAction again?	4.8

To improve ActiveAction stakeholders have expressed a number of suggestions: to split the Intensive-Day into two sessions; add ambient music; reduce the duration of breaks but increase their number and make a summary at the end of each practice, mainly in Buy a Feature practice.

Consultants have observed that: information about stakeholders is fundamental to prepare Intensive-Day; a list of pre-prepared questions about the stakeholders' business is useful to "re-activate" discussions; more "standing-up" practices should be added; playing with LEGOs is not fun for everybody; AVAXes could be created and reviewed in teams instead of individually; Non-Functional Requirements practice could be omitted if the stakeholders do not have technical background; and lastly, if the product owners are not present Intensive-Day must be postponed.

6 Conclusions and Future Work

Considering the non-technical nature of software project Inception activities, stakeholders can be involved in depth and join software developer organization at this particular moment. In fact, addressing factors like executive support, engagement of key stakeholders and clearly defined needs increases the success rate of a project. However, stakeholders' involvement is a complex action that requires investment of extra effort and time and be intrusive for both parties.

The strategy of using games within software developing projects in order to address and improve collaboration between stakeholders and development teams can be considered as a novel and beneficial approach. Consequently, game-based approach has been adopted as a strategy to improve the Inception phase outcomes.

ActiveAction is a workshop service which objective is to clarify and process stakeholders' needs and priorities. During the Intensive-Day, game-based activities provide a favorable environment in which stakeholders and development teams collaborate and produce results of great importance for the Inception phase of a project.

Two lines are identified as future work: (i) to replicate the workshop in affiliates and (ii) to define a continuous improvement process for ActiveAction. At the moment one affiliate has replicated its first three ActiveActions with promising results. Moreover, documenting the method will reduce the possibility of variation and find ways to improve.

Finally we can conclude that the inclusion of games in such a challenging and over-whelming activity as software projects inception, is feasible and reported promising results which benefit both stakeholders and software developer organizations.

Acknowledgements. The authors thank Miriam Eréndira Jiménez-Hernández and Roberto Azrael Medina-Díaz for on-site observation and initial analysis of the data.

This work has been funded by the GEODAS-BC project (Ministerio de Economía y Competitividad and Fondo Europeo de Desarrollo Regional (FEDER), TIN2012-37493-C03-01); GLOBALIA project (Consejería de Educación, Ciencia y Cultura (Junta de Comunidades de Castilla — La Mancha) and FEDER, PEII11-0291-5274); SDGear project (TSI-100104-2014-4), framed under the ITEA 2 Call 7, and co-funded by "Ministerio de Industria, Energía y Turismo (Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica 2013-2016) and FEDER"; the Graduate Science and Engineering Computing (PCIC-UNAM) and the grant scholarship program of CONACYT.

References

1. Michael, D., Chen, S.: Serious Games: Games That Educate, Train, and Inform, 1st edn. Muska & Lipman/Premier-Trade, Cincinnati (2005)
2. Abt, C.: Serious Games. Viking Press, New York (1970)
3. Serious game initiative. http://www.seriousgames.org/
4. The Standish Group International, Inc., CHAOS Summary for 2010 (2010)
5. Stepanek, G.: Software Project Secrets: Why Software Projects Fail, 1st edn. Apress, New York (2005)
6. Kruchten, P.: Rational Unified Process – An Introduction, 3rd edn. Addison-Wesley, Boston (2003)
7. Gartner survey shows why projects fail. http://thisiswhatgoodlookslike.com/2012/06/10/gartner-survey-shows-why-projects-fail/
8. Ambler, S.: Defining success. Dr. Dobb's J. (2007). http://www.drdobbs.com/architecture-and-design/defining-success/202800777
9. Ibrahim, R., Ayazi, E., Nasrmalek, S., Nakhat, S.: An investigation of critical failure factors in information technology projects. J. Bus. Manage. IOSR **10**(3), 87–92 (2013). ISSN 2319-7668
10. Geneca, LLC: Doomed from the start? Why a majority of business and IT teams anticipate their software development projects will fail. Geneca Industry Survey (2011)
11. Bloch, M., Blumberg, S., Laartz, J.: Delivering Large-Scale IT Projects on Time, on Budget, and on Value. Insights & Publications, McKinsey & Company, New York (2012)
12. Mountain goat. http://www.mountaingoatsoftware.com/scrum/product-owner/
13. Passos, E., Medeiros, D., Neto, P., Clua, E.: Turning real-world software development into a game. In: Proceedings of the Brazilian Symposium on Games and Digital Entertainment, pp. 260–269 (2011). ISBN: 978-0-7695-4648-3
14. Claypool, K., Claypool, M.: Teaching software engineering through game design. In: Proceedings of the Innovation and Technology in Computer Science Education Conference, vol. 37, no. 3, pp. 123–127 (2005). ISBN: 1-59593-024-8
15. Gamestorming. http://www.gogamestorm.com/
16. Innovation games. http://innovationgames.com/
17. Knauss, E., Schneider, K., Stapel K.: A game for taking requirements engineering more seriously. In: Proceedings of the International Workshop on Multimedia and Enjoyable Requirements Engineering, pp. 22–26 (2008). ISBN: 978-0-7695-3626-2
18. Torvinen, V.: The labour game method. In: Proceedings of the International Workshop on Database and Expert Systems Applications, pp. 382–386 (1999). ISBN: 0-7695-0281-4
19. Entia. http://www.entia.com.mx/
20. Morales, M., Oktaba, H.: KUALI-BEH kernel extension. Annex B (informative). In: Essence – Kernel and Language for Software Engineering Methods. Object Management Group (2012)
21. Hohmann, L.: Innovation Games: Creating Breakthrough Products through Collaborative Play, 1st edn. Addison-Wesley Professional, Boston (2006)
22. LEGO: LEGO Serious Play: Open Source. The LEGO Group, Billund (2010)
23. Cantoni, L., Faré, M., Frick, E.: User Requirements with LEGO. University of Lugano. webatelier.net & NewMinE Lab, version 1.0 (2011)
24. Osborn, A.: Applied Imagination: Principles and Procedures of Creative Problem Solving, 3rd edn. Charles Scribner's Sons, New York (1993)