The Journal of Systems and Software 132 (2017) 21–40

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

A framework for gamification in software engineering

Félix García[a,*], Oscar Pedreira[b], Mario Piattini[a], Ana Cerdeira-Pena[b], Miguel Penabad[b]

[a] Institute of Technology and Information Systems, University of Castilla-La Mancha, Ciudad Real, Spain
[b] Database Laboratory, University of A Coruña, A Coruña, Spain

ARTICLE INFO

Article history:
Received 23 September 2016
Revised 24 March 2017
Accepted 8 June 2017
Available online 15 June 2017

Keywords:
Gamification
Software engineering
Methodology
Ontology

ABSTRACT

Gamification seeks for improvement of the user's engagement, motivation, and performance when carrying out a certain task; it does so by incorporating game mechanics and elements, thus making that task more attractive. The application of gamification in Software Engineering can be promising; software projects can be organized as a set of challenges which can be ordered and that need to be fulfilled, for which some skills, and mainly much collective effort, are required. The objective of this paper is to propose a complete framework for the introduction of gamification in software engineering environments. This framework is composed of an ontology, a methodology guiding the process, and a support gamification engine. We carried out a case study in which the proposed framework was applied in a real company. In this project the company used the framework to gamify the areas of project management, requirements management, and testing. As a result, the methodology has clearly enabled the company to introduce gamification in its work environment, achieving a quality solution with appropriate design and development effort. The support tool allowed the company to gamify its current tools very easily.

1. Introduction

The software engineering field has been experiencing a positive evolution over the last few years. Over the last decades, the main attention has been focused on improving software processes by using standards such as CMMI (CMMi for Development; ISO/IEC 2003) or ISO 15504 (ISO/IEC 2003; ISO/IEC 2008), incorporating agile development (mainly SCRUM), or promoting product quality improvement (ISO 25010), among other measures. But the main asset in software development as compared to other disciplines (manufacturing, industrial processes) is the human factor whose motivation and engagement are keys to success. People management in software projects has been considered a key issue (DeMarco and Lister, 2013); this is the case in reference models such as People CMM (Curtis et al., 2001), Personal Software Process (Humphrey, 2000) or Team Software Process (Humphrey et al., 2010). However, despite all these efforts, the nature of software engineering tasks, which are usually tedious, is a factor which poses a threat to engagement and motivation, both on the part of project managers and on that of team members.

To foster this engagement, it is important to consider that software projects can be organized as a set of challenges which can be ordered and that need to be fulfilled, for which some skills, and mainly much collective effort, are required. Software processes can involve discipline and agility; software creativity also shows itself to be a rational approach for combining the best of these qualities (Glass, 2006). An important similarity between software engineering and a game can therefore be detected: they involve activities in which a player learns new skills and uses and combines these to achieve certain challenges, obtains rewards or receives punishments, depending on success or failure, respectively (Passos et al., 2011).

Given this context, the application of gamification in SE seems to be promising. Gamification is widely recognized as "the use of game design elements in non-game contexts" (Deterding et al., 2011). Gamification uses the philosophy, elements, and mechanics of game design in non-game environments to induce certain behavior in people, as well as to improve their motivation and engagement in a particular task. That is to say, gamification takes those features that make real games fun and attractive (and even addictive), and uses them to improve the player experience in a non-game environment, such as the workplace, the school, a software application, or customer-oriented web site. This field has experienced significant growth and popularity in the last few years (Deterding et al., 2011; Zicherman and Cunningham, 2011; Werbach and Hunter, 2012; McMillan, 2011).

In the SE field, therefore, researchers and practitioners are not unaware of the potential benefits of gamification in the workplace. Software development has fun aspects, but most software

* Corresponding author.
E-mail addresses: felix.garcia@uclm.es (F. García), opedreira@udc.es (O. Pedreira), mario.piattini@uclm.es (M. Piattini), acerdeira@udc.es (A. Cerdeira-Pena), penabad@udc.es (M. Penabad).

http://dx.doi.org/10.1016/j.jss.2017.06.021
0164-1212/

developers find certain tasks boring and tedious. Examples include coding parts of the system that are repetitive (such as the data access code, that is very similar for all domain objects), writing unit tests for not-challenging functionalities, or fixing tons of easy maintenance issues, among others. Gamification allows the organizations to explicitly reward developers for every aspect of their work, for every task they complete, for every unit test they write, and for every issue they solve. The mechanics of gamification not only give us a way to reward people, but they may even make work funnier. Social elements such as collaborative quests or rankings in which developers or teams compare their progress with that of their partners are some examples of this. In this way, they continuously receive feedback on the value of their work for the project and for the organization. Therefore, introducing gamification in the daily work of software engineers can help software development organizations to improve motivation and engagement of developers in their work and, therefore, the results of software projects, both in terms of product quality and project performance. Although this work mainly focuses on the company and the developers that participate in the process, the benefits of incorporating gamification to SE concern both the company as an organization, the employees and the customers.

The incorporation of game mechanisms in tools supporting software development has therefore emerged as a current trend (Passos et al., 2011). The main gamification elements which are essential in such tools are: game mechanics applied in a cycle of challenges-rewards-punishments; badges or medals, which are obtained as a result of achievements, and feedback of performance of individuals and teams as compared to others. Existing commercial tools are also starting to support SE processes by incorporating the basic gamification mechanisms mentioned; these include, for instance, *Gamify*,[1] *RedCritter*,[2] *PropsToYou*,[3] *ScrumKnowsy*,[4] *Masterbranch*,[5] *CodeHunt*[6] or *Jenkings Continuous Integration game plugin*,[7] among others.

But application of gamification in SE is scarce, and the little research that does exist on gamification applied to SE is very preliminary, or immature. Many existing pieces of work consider only the basic gamification elements, and there is no systematic methodology to incorporate gamification in SE and thus improve user engagement and performance. In addition, the incorporation of gamification has not been done properly, as it is often integrated into an independent tool, usually developed ad-hoc for this purpose (ISO, 2015).

It is therefore fundamental to provide software organiations with additional support to adopt gamification in their processes, which must adhere to special and specific characteristics of software development and management. To achieve this objective, in this paper we propose the GOAL (Gamification focused On Application Lifecycle Management) framework; it is intended to support the integration of gamification in software engineering environments. The framework aims to provide software companies with the conceptual basis and the corresponding technological support to apply gamification, while also supplying suitable guidelines and solutions for integrating gamification into the organization's tool ecosystem. In order to obtain initial feedback about the suitability of the framework in a real setting, the proposal has been applied in a software company from Spain.

The rest of this paper proceeds as follows: in the following section the state of the art as regards gamification in SE is analyzed. Section 3 presents the framework by describing its three main elements (ontology, methodology and tool). In Section 4, a case study of the application of the GOAL framework is presented. Finally, the conclusions of this work are set out and future research is outlined.

2. Related work

The field of gamification is a vast research area that has attracted significant attention in various application domains, such as online marketing, software apps for mobile devices, or education, among others (Zicherman and Cunningham, 2011; Hugos, 2012). One of the most significant lines of research in this field has been the evidence about the usefulness of gamification; this was initially evaluated by Hamari et al. (2014) by means of a literature review. As a result, it was concluded that "gamification does work, but some caveats exist" ; most papers report positive results from gamification (with some empirical evidence), but some underlying confounding factors were also present. Gamification in web applications was analyzed in the literature review of Xu (2012), concluding that gamification was based on superficial game mechanics (point, level, leaderboard and badges) and that more advanced aspects should be considered. Those might include features such as social interaction, mobility (by supporting the ubiquity of mobile devices), and analytics, which must be enhanced.

Over the last few years, much research work has focused on the introduction of gamification in software engineering environments. In previous work, we conducted a systematic mapping of the existing research in the field of gamification applied to software engineering (ISO, 2015), whose main goals were: to identify which software processes had been considered in existing research; what gamification elements and mechanics have been applied; the research methods followed; and the type of forums in which those primary studies were published. The mapping considered both academic and non-academic publishing sources when searching for primary studies. Other relevant questions addressed in the systematic mapping were the type of methodology applied in the gamification, the extent to which the gamification was integrated with the organization's tools, the empirical evidence on the impact of gamification in the participant's engagement, on motivation, and on the organization's results. As a result, we found that the first primary studies date back only to the year 2010, so this research line is quite young. The 29 primary studies found were published up to June 2014; this is also a low number of relevant papers. The main findings extracted were:

- Software implementation was the software process that has attracted most interest to date, followed by software testing and requirements management. Other important process areas, such as those related to project management, were also considered, but in very few primary studies.
- Regarding the type of game elements and mechanics applied in existing proposals, point-based reward systems and badges were the most relevant, followed by rankings, social reputation elements, and dashboards.
- Half of the primary studies were philosophical papers about the field, or papers presenting a proposal, but without any kind of real application to validate its appropriateness. Fewer papers considered an empirical evaluation or validation of their proposals in real settings, and many works that did so carried them out in academic environments.

Other questions analyzed were: the methodology followed in the application of gamification, the degree to which the proposals

[1] https://marketplace.atlassian.com/plugins/com.sentinel.confluence.plugin.gamify-core/server/overview.
[2] https://www.redcritter.com/.
[3] http://propstoyou.com/.
[4] http://www.scrumknowsy.com/.
[5] https://www.masterbranch.com/.
[6] https://www.codehunt.com/.
[7] https://wiki.jenkins-ci.org/display/JENKINS/The+Continuous+Integration+Game+plugin.

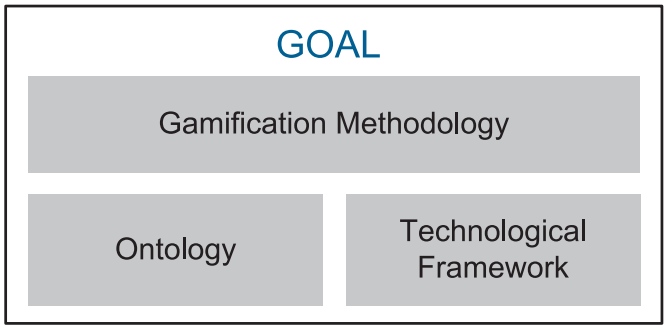

Fig. 1. GOAL Components.

were, or could be, appropriately integrated with the tools of a real organization, as well as evidence on the effect of gamification in the motivation of the participants and the results of the organization, and the replicability of the proposals in other organizations. We found relevant findings which allowed us to reach the conclusion that the application of gamification to software engineering is at a very early stage and is not fully mature. We observed that the adoption or gamification in SE is going more slowly than in other domains such as marketing, education, health, or banking. Two main weaknesses were identified:

(1) The lack of a sound methodological support for the application of gamification, which directly affects the replicability of the proposals in other organizations or scenarios. In many cases, the gamification proposal was applied ad-hoc. In those cases in which the authors outlined a series of steps for applying it, these were usually too simple and at times even incomplete.
(2) The difficulty of integrating the gamification proposals in real workplaces, which is an even more important issue than the former one. In those cases in which the proposal was implemented, that implementation was usually done through a custom application, which can be very limiting. The reason is that in any company, but especially in those companies with a large team of engineers, the adoption of a CASE or other support tool is neither easy to decide, nor easy to deploy in the organization. This means that the chances that corporate software (for requirements management, project management, or testing, to name but a few instances) will be replaced by a gamified tool are very small; even if they are gamified, these tools do not usually even get close to existing professional CASE tools at a functional level.

After updating the former systematic mapping by considering related works on gamification in software engineering including those published in 2015 and 2016, we can assert that the conclusions of the systematic mapping are still valid as of today.

According to the aforementioned issues, the aim of this paper is to fill that gap by means of GOAL (Gamification On Application Lifecycle management), a complete framework for gamification in software engineering domains, which is described in the next section.

3. GOAL: gamification focused on application lifecycle management

The GOAL framework aims to support gamification in software engineering activities by providing the required conceptual, methodological and technological support. The framework is composed of three main elements (see Fig. 1). A gamification ontology has been developed to promote knowledge-sharing and application about gamification in software projects. We also present a methodology to guide the gamification of software lifecycle or specific activities; it guides the gamification from the target business objectives to the design of a gamification solution and the evaluation of its effectiveness. Finally, we have designed a generic software gamification architecture from which a software engine has been built. This engine can be integrated in the software engineering platforms of the companies to support gamification in their software projects (see Fig. 7). The main aim of this architecture is to integrate the existing workplace tools with gamification supported by the software engine that would centralize the evaluation of the behavior of the participants, rewarding them as stated by a set of game rules. Notice that although we propose these three elements as components of an integrated gamification framework, each of them can be used independently. The elements are described in the following subsections.

3.1. GOAL conceptual support: an ontology for SE Gamification

The first main step in the development of the framework was to characterize the gamification domain in SE by identifying the relevant concepts and their relationships by means of ontology. To build the ontology, we extracted the concepts from the primary papers obtained in the systematic mapping (ISO,), as well as from the reference books about gamification (Zicherman and Cunningham, 2011; Hugos, 2012) and the main conferences. Fig. 2 shows the overall ontology developed, represented by using a UML class diagram.

Namely, the ontology aims to capture the relevant conceptualization grouped in these main aspects:

3.1.1. SE Gamification scope and goals

A gamified system evaluates the interaction of software practitioners with their *SE tools*, along with their performance and the quality of their work according to a set of defined goals (Fig. 3). A ***game*** is a structured playing and a ***SE tool*** is defined in this context as accessories that help developers develop/maintain software programs or manage theirs more effectively. A SE tool can be gamified with several ***games*** and a game can involve gamifying several SE tools. A ***goal*** represents a challenging target to be achieved with gamification, one which can be evaluated by means of fulfilling conditions. Example of goals can be: to improve employee motivation; to increase productivity, etc. Practitioners (***employees***) can assume one or several roles in the ***game***; for each one a ***player*** is created. Players can be ***individual*** or organized in ***teams***. An individual player can belong to one or several teams. In addition, players can have some social interaction with other players (relationship “has friends”). Moreover, a player can be categorized according to a ***PlayerType***, which includes the types of players who can be found in a game. A categorization can be for instance the proposed by Bartle’s (Bartle, 1996), which comprises: ***Killers***, who like to provoke and cause drama and/or impose them over other players in the scope provided by game; ***Achievers,*** who are competitive and enjoy beating difficult challenges whether they are set by the game or by themselves; ***Explorers***, who like to explore the game and to discover the finer details of the game mechanics; and ***Socializers***, who are usually involved in the community aspect of the game and they can be more interested in having relations with the other players than playing the game itself.

3.1.2. SE Gamification: mechanics and aesthetics

Fig. 4 shows this part of the ontology. With regards to **mechanics**, their related concepts must represent the basic actions and control mechanisms and processes that are necessary to gamify the interaction of practitioners with SE tools, by including the rules that govern those actions. This is represented by the fact that a game is composed of a set of ***evaluable actions*** (abstract

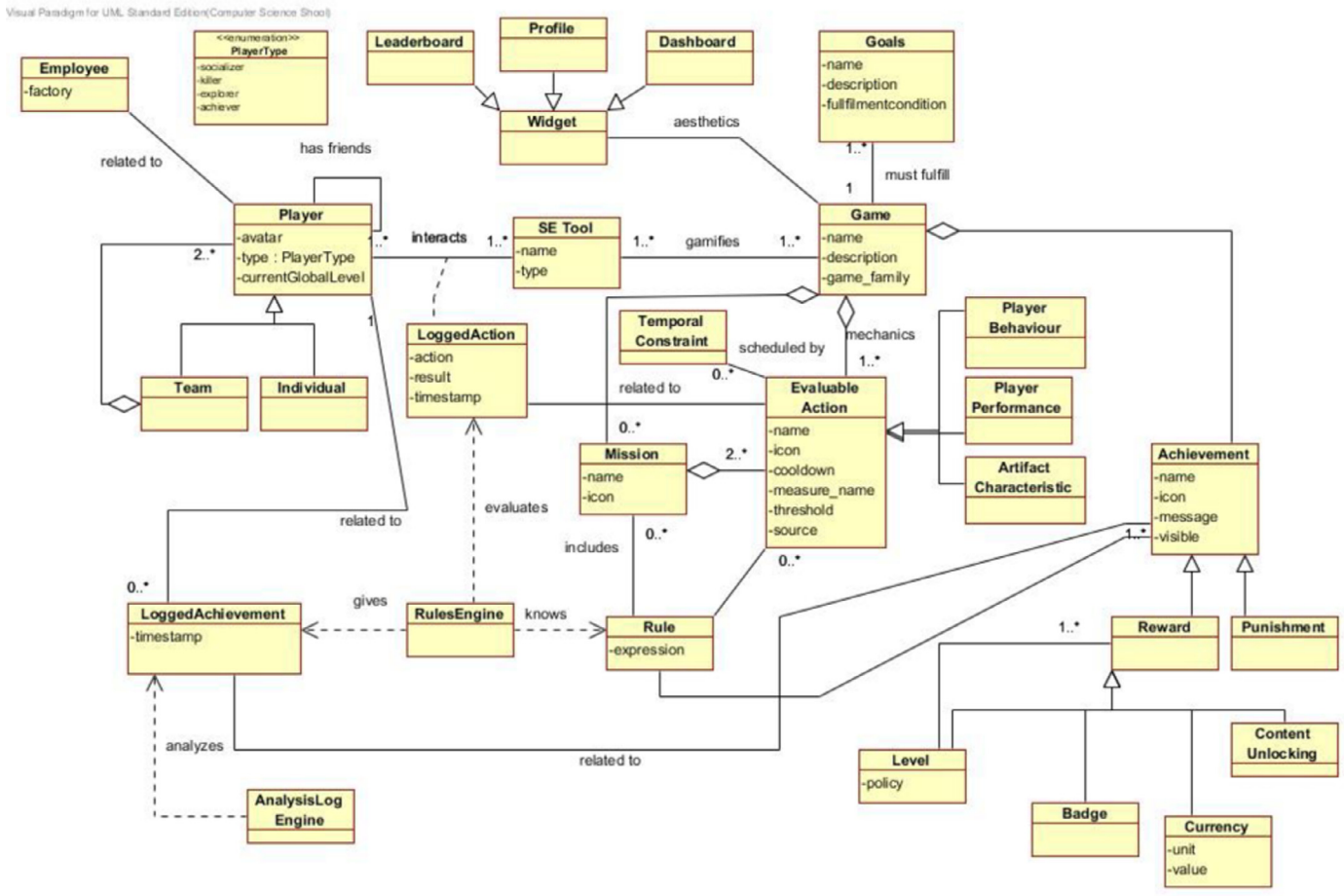

Fig. 2. GOAL ontology.

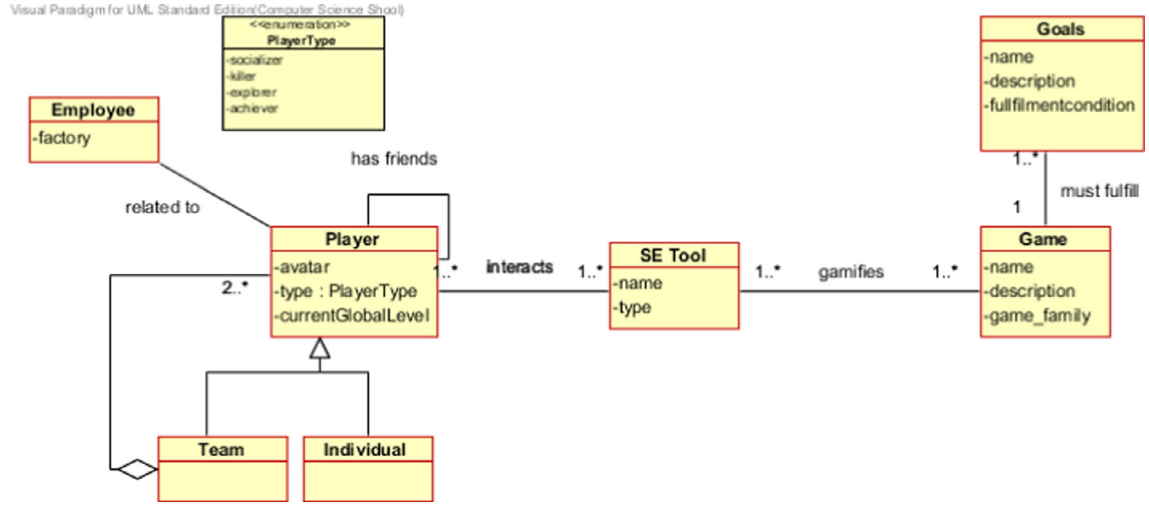

Fig. 3. GOAL ontology: SE Gamification scope and goals.

concept), which can be organized into ***missions***, according to certain ***rules***. An evaluable action is an action or condition related to ***players*** which can be evaluated in the game to assess if an achievement can be obtained as a result of it. These actions can be: ***player behavior***, which means the direct interaction of the player with the SE tool or platform (for instance a new commit in a configuration management system or a new solved issue in a ticketing system); ***player performance***, which is a measurable characteristic of the player, related to her/his performance in the game; and ***artifact characteristic***, which is a measurable characteristic of an artifact produced as a result of the interaction of the player with the SE Tool, as for instance, the maintainability of the code uploaded by the player. Evaluable actions can be limited by ***temporal constraints***, as for example, a "post review" evaluable

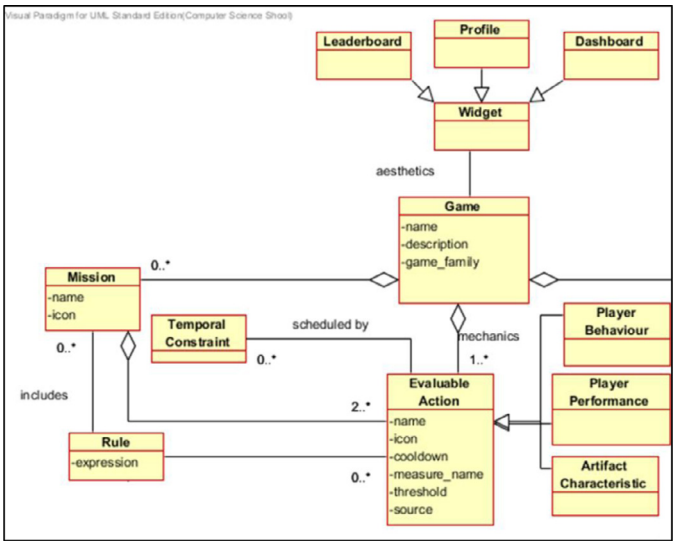

Fig. 4. GOAL ontology: Game Mechanics and Aesthetics.

action is only evaluated if the post has been produced before 22nd Nov 2015. A ***mission*** (or quest) is a "journey of obstacles" (evaluated by evaluable actions) a player must overcome: Missions are a way to organize player effort.

On the other hand, the **aesthetics** of the ***game***, intended as the visual representation of the game players and their results, is represented by the following concepts: A ***Widget***, is an additional feature or extension to a software program, operating system GUI, or web page. These give the software or web page additional features and capabilities that help give the user exactly what he or she wants and provides an overall better experience; User's data about their results in the game are represented in ***Profiles*** which usually displays the general information of the player: *avatar; achievements*, as a representation of their collected currencies, badges, levels and progression; *activity feeds* which show players what has been taking place in the system as a whole, and motivates the player to obtain the same achievement as others. The former representations are completed by means *of leaderboards and dashboards.* A ***leaderboard*** is a means by which users can track their performance (compared to others), and visually display where a user stands in regards to other users. Leaderboards can be broken down into several subcategories, such as: Global, Friends, Relative, Isolated etc. A ***dashboard*** is a customizable visualization of player results, and other player information.

3.1.3. SE Gamification: dynamics

In Fig. 5 the conceptualization related to the evaluation of players are depicted. These concepts are related to the satisfaction of player desires, also known as dynamics. Namely, as a result of the evaluation of the actions (***logged action***) by the ***rules engine***, an ***achievement*** can be obtained and logged. A ***logged action*** is an action which is produced as a result of the interaction of the player with the SE tool which is registered to be evaluated. It represents the "execution time" concept of player behavior. An ***achievement*** represents a ***reward*** (positive evaluation) or ***punishment*** (negative evaluation) which is obtained by the player as a result of the evaluation of actions or mission according to game rules. Several types of rewards can be considered, such as: ***Content Unlocking***, which consists of making contents which are hidden by default visible to the player; ***Level***, which is a position or status in a scale of values which means the amount or degree of progress in the game; ***Badge***, which is a distinguishing emblem or mark which represents an achievement and; ***Currency***, which is a reward of a numerical nature which is accumulated throughout the course of the game. Achievements of the players must be registered by the gamified system, for which it is important to consider the ***logged achievement,*** which represents the achievement which is obtained by the players during the game and it is logged into the system. In addition, the logging and evaluation of the achievements must be supported, namely by: the ***RulesEngine,*** which is responsible for observing logged actions, as a result of which it checks the rules which are related to action. And thanks to rule-checking, the engine can create and log achievements; and the ***Analysis Log Engine,*** which checks if a set of related achievements can produce a higher level of achievement (for instance a level achievement), wherever this new achievement is logged by the engine.

By means of this ontology, a common vocabulary is provided to practitioners, making it easier for them in their tasks of knowledge-management and sharing. With this, the different ingredients of the gamification are characterized, but the most important thing is to support as well as possible the creation of good recipes (games) which combine these ingredients (game mechanics, dynamics and aesthetics) in the best way to satisfy the stated goals.

3.2. GOAL methodological support

The aim of this methodology is to assist SE gamification designers in the development of gamified systems which improve user engagement and performance. This process of gamifying a work environment should follow a series of guided steps which follow clearly- identified business goals. The main desirable characteristics of the methodology can be summarized in:

- Player-focused. The players are the main actors and have the greater impact in the benefits of the gamified system. The key impact of the human factor in software engineering projects is well-known, hence the importance of suitably managing their main motivations (which can be different at each step) to obtain their best performance.
- Balanced Analytics and Creativity. The design of games has a high creativity load, but software development implies the application of certain techniques, (which needs some rigor) throughout the whole software lifecycle.
- Iterative and prototype-based, as the development of gamified systems can require several iterations and some agility in many cases, due to frequent changes. The building of prototypes is also an essential aspect in managing the risks which can arise during gamified system development.

Bearing all this in mind, the GOAL methodology intends to cover all the stages of gamification applied to SE in a thorough way, by combining the best of existing methods and techniques about the two main fields to consider: gamification and software development (El-Nasr et al., 2005; Kumar et al., 2013; Kumar and Herger, 2013; Keith, 2010). In order to build the methodology, a rigorous review of relevant methods and techniques was carried out, so as to satisfy the desirable characteristics mentioned above, to identify the activities which require to be supported (from goals specification to continuous measurement and improvement of the gamified platform) and the techniques which can be applied in each stage. In addition, the methodology was intended to provide practitioners with some flexibility for it to be tailored according to their company and project context.

Table 1 outlines the GOAL methodology activities with their corresponding tasks, along with the main sources which inspire the supporting techniques and artifacts for each activity:

Table 1
GOAL Methodology.

Id	Activity	Description	Relevant Sources
1	**Identify the objectives of the gamification**	The objectives of the gamification program are defined, along with the corresponding indicators which must assess the fulfillment of objectives.	(Kumar and Herger, 2013)
	Tasks: 1.1 Define the current scenario 1.2. Define the target scenario 1.3. Establish the SMART mission		
2	**Player Analysis**	Each player is analyzed in her/his context in order to align player profiles with gamification objectives.	(Kumar and Herger, 2013) Players can also be categorized according to Bartle's player types (R. Bartle) or -Marczewski's Gamification User Types (Marczewski, 2014).
	Tasks: 2.1. Identify the organization culture and types of players 2.2. Collect the demographic and psychographic information of all the players (including type) and add this to their profile.		
3	**Gamification Scope Definition and Feasibility Study**	To define the scope of the gamification (software lifecycle coverage and intrinsic/extrinsic motivators of the game) and to conduct a feasibility study (economic, technical, operational and legal), in order to choose the best solution.	(Burke, 2014) See also (Zicherman and Cunningham, 2011), for example, for discussion on the role of intrinsic and extrinsic motivation in gamification.
	Tasks: 3.1. To establish the scope, motivators, type of game (collaborative, competitive, individual) and different possible solutions. 3.2. To conduct feasibility analysis for each alternative and to choose the solution		
4	**Game Analysis and Design**	The main result of this activity is the set of requirements (game use cases) of the software tool which support the game, and automatize the game and its elements.	MDA framework Gamification Model Canvas [26]
	Tasks: 4.1. Choose the game components 4.2. Choose the game mechanics 4.3. Establish the game economy 4.4. Establish the game dynamics (rules) 4.5. Establish the game aesthetics 4.6. Elaborate the game use cases		
5	**Development of the Gamified SE Platform**	Software development of the gamified platform for SE from the stated game use cases.	An agile approach is suggested, driven by game use cases; it combines management principles of Scrum and Kanban (Keith, 2010); (Jacobson et al., 2011) with specific techniques of videogame development (El-Nasr et al., 2005).
	Tasks: For each Sprint: 5.1. Sprint Management 5.2. Sprint Development (Preparation, Development (Analysis, Design, Implementation, Asset Creation), Testing (debugging, correction and improvement, optimization)		
6	**Managing, Monitoring, Measuring**	The gamified platform is periodically monitored to analyze the performance and achievement of the business objectives. When deviations are detected, action plans are developed to refine or include in the game the elements required (components, mechanics, dynamics, aesthetics)	(El-Nasr et al., 2013)
	Tasks: 6.1.Collect indicator values from execution logs 6.2 Analyze the indicators and evaluate the fulfillment of business objectives 6.3. Develop action plans to improve the gamified system		

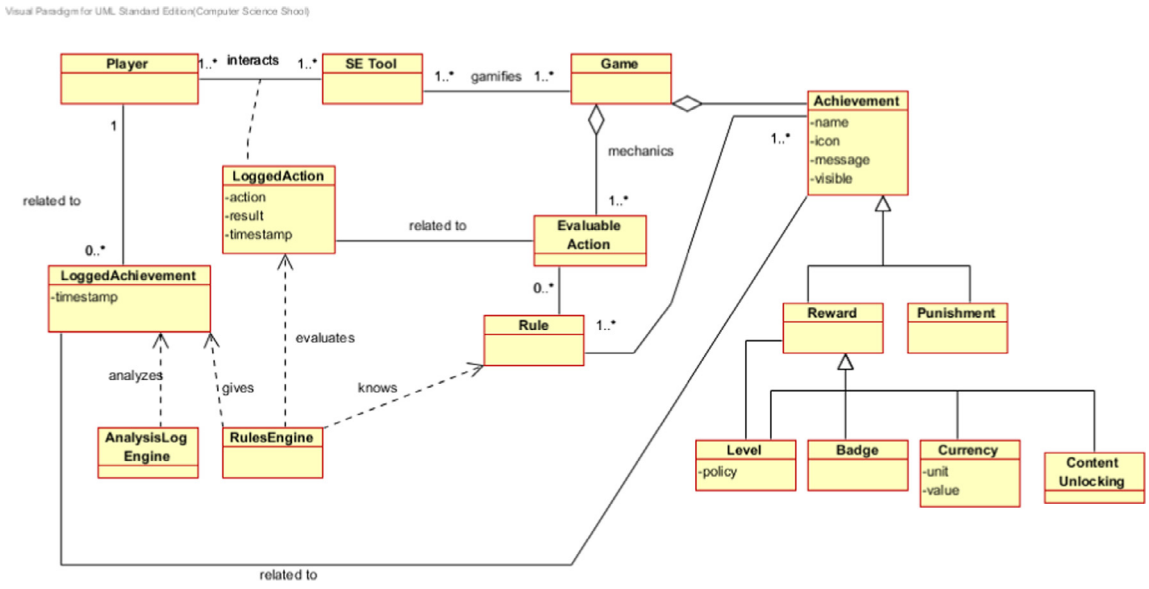

Fig. 5. GOAL ontology: Game Dynamics.

3.2.1. Identify the objectives of the gamification

The purpose of the first step of the methodology is to establish a set of goals for the gamification program; these should be aligned with the business objectives. To analyze these gamification goals we must examine the current and desired scenarios; that is, we need to see which situations could be improved in some way by means of the application of gamification, and how they could be improved. We must then translate these analyses into SMART goals, in other words, into objectives that are Specific, Measurable, Actionable, Realistic, and Time-bound. In measuring the real improvement in each goal, we should identify possible metrics and indicators, quantitative whenever possible. Notice, however that this may not be possible for all goals or for organizations with non-existent or immature measuring and analysis activities. Nonetheless, if the measuring system of the organization allows it, the business objectives could even be established in terms of an objective value of the indicators involved. For example, an organization could establish a business objective consisting of raising the productivity of the development tasks to below 90% (that is, on average, a development task requires less than the 90% of the effort estimated to complete that task).

In Appendix A.1 we present a template for documenting the different business objectives identified by the organization; it specifies the current and desired scenarios for each business objective, along with the quantitative/qualitative indicators and objective values associated with each of them.

3.2.2. Player analysis

As pointed out by many authors on general gamification sources (Deterding et al., 2011; Zicherman and Cunningham, 2011; Hugos, 2012), analyzing the players for which we are designing the gamified environment is very important if we are to end up with a solution that is able to improve their motivation and engagement in the tasks in hand. To carry out this analysis, we could follow the approach known as “player centered design”, proposed by Kumar and Herger (Kumar and Herger, 2013), in which the players, along with their motivations, mission, and mechanics, are the key element in the gamified environment.

Player analysis can be divided into two subtasks. First of all, we should identify the organizational culture and the player types (which can correspond directly with some roles within the organization). The organizational culture can be described in terms of, for example, characteristics such as its working system, processes, and maturity, the level of education and training of its employees, as well as work environment and organizational structure, or aggregated employee characteristics, such as age.

The player types can be characterized according to the proposals of, for example, Bartle’s player types (Bartle, 1996) (killers, achievers, socializers, explorers) as presented in Section 3.1, or Marczewski’s gamification user types (Marczewski, 2014). Of course, a player does not usually fall completely into just one of these categories. This classification can also help the designer in deciding which gamification mechanics to incorporate to the solution, since Bartle associates typical mechanics of interest to each player type. Marczewski (Marczewski, 2014) proposes other classifications of player types. These classifications of player types can be used alone in combination. The description of the players can be completed by gathering psychographic and demographic data and adding this to the profile descriptions.

In Appendix A.1 we present two templates that can be used in the player analysis. The first one allows for an individual analysis of each player, including demographic characteristics (such as the age, gender, education level, work experience, job position in the company, etc.), and psychographic characteristics (which might include interests, opinions, values, etc.). As usual, this analysis can be performed for all players, or for a sample of players if the total number of players is too high. A second template allows the categories of the players to be summarized according to Bartle’s taxonomy.

For some organizations, it may be useful to rely on user tests for player analysis. That is, organization could prepare tailored questionnaires, according to the templates provided Appendix 1, for their employees in order to identify characteristics such as interests, opinions, and values. As a matter of fact, the choice of the specific method to be used for player analysis depends on the size and the capacity of the organization. In any case, the analysis of players should take into account both the current employees of the company and future employees, since the personnel rotation in software development companies can be high. A way of including future employees in the analysis is to analyze an abstract player from the ideal profile, requirements, and data of the human resources function in the organization.

3.2.3. Define the scope of gamification and alternative solutions

The information obtained in the analyses carried out in the previous two tasks allows us to define an initial approach to the gamification solution. We must first identify the scope of the solution, including which processes, user types, and tools will be affected. The entry for the definition of the scope comes mainly from the gamification goals established in the first task of the methodology. From the information about the players we obtained in the second task, we can then decide what type of gamified environment we want to create, and determine a first alternative (or more alternatives if that is the case). Burke's gamification scenarios (Burke, 2014) are a valuable source for this task.

In some cases, it may be useful to formulate the alternative solutions for the gamification by making an analogy with a traditional game. For example, if a company wants to gamify its maintenance process by rewarding developers upon solving maintenance issues that arrive continuously, a possible solution to this problem can be formulated by making an analogy to the traditional space shooter game, that is, the developer would play the role of the space shooter, and the issues that arrive to the projects are the targets he/she has to shoot (and those issues with a higher priority can be seen as enemies that come faster).

It is also important to carry out a feasibility analysis for the alternative that has been outlined at this step; this would consider economic, technical, legal and operational aspects.

3.2.4. Game analysis and design

The previous three tasks focus on performing a preliminary analysis of the goals, the participants, and the scope of the gamified environment. This task directly addresses the analysis and design of the gamified environment. This analysis and design will include which actions we want to reward, what types of rewards we want to grant to the players (gamification components and mechanics), and how the gamification rules will associate rewards with each action (gamification economy and dynamics); it will also include the environment through which we will present the participants with all this information (aesthetics), along with the use cases or scenarios through which they will interact with the gamification environment. This may therefore be the most important task of the process, since it is here that we will design the specific game rules and elements that the users will interact with.

The first step in the analysis and the design of the gamified environment should be to decide what mechanics will form the base of the game. This decision would be based on the result of the player analysis. If the analysis of players is carried out following the frameworks of Bartle (Bartle, 1996) or Marczewski (Marczewski, 2014), those systems already provide a hint as to what game mechanics could work best for each player prototype. Dignan (Dignan, 2011) proposed a frame composed of game construction blocks, which provide a valuable source for this task.

We must then design the game economics; that is, we must determine the types of rewards the players can obtain from their actions. This point offers lots of possibilities. Kumar (Kumar and Herger, 2013) provides a valuable framework for this task. As pointed out in (Kumar and Herger, 2013), the rewards obtained from players' actions can be classified into four categories, namely self-esteem (leadership, conquest, mastery, access, praise), fun (discovery, excitement, awe, delight, fantasy, surprise), social capital (likes from other players, friends, contribute, charity, groups, status), or things (points, cash, resources, rewards, prices). The most widely-used game mechanics applied in software engineering (see (ISO,)) are a combination of points, levels, and badges. Points act like a monetary reward for each action, since we can modulate the number of points rewarded to the player, based on the importance of each action. Regarding points, levels provide the players with a status that can be compared directly with that of their workmates; this can foster motivation and competitiveness. Badges usually complete these set of game mechanics by providing a way of rewarding significant milestones in the activities of players. However, many other game mechanics have been studied and can be applied (for more information on this aspect, see (Zicherman and Cunningham, 2011; Werbach and Hunter, 2012; Hugos, 2012)). Game economics does not only have to involve choosing how actions are transformed into points or badges. In some cases, companies link points from the gamification environment with real-life rewards, such as money, or virtual money that can be used to acquire things such as training courses related to the company's activity, or time for personal projects, for example.

The link between players' actions and rewards is defined by the rules of the game, which should evaluate each action from a player, and determine if that action deserves a reward; in the case of quantitative rewards, it should establish the amount. An example of a game rule could be the following: "when developers complete a development task, if the effort (in hours) they devoted to that task is smaller than or equal to the estimated effort, then the developer will receive a number of points equal to the number of hours estimated for that task".

The rules of the gamified environment define the economics of the gamified environment, which must be designed with care to avoid undesirable situations, since the chances of cheating are a relevant aspect of the game economics. In the example of game rule we have just presented (rewarding tasks completed with an effort below what was estimated), this rule could tempt some players to artificially inflate their effort estimations for the development tasks, so that they can always complete them within that estimated effort, and therefore obtain a significant reward. It could also tempt the players to complete the tasks in less time, but at the same time with less quality. This is a common problem in gamification economics (not only in the case of SE), and the design of the rules of the game must take these situations into account. In the case study presented in the next section, a real company dealt with this particular problem. In their case, the effort estimation of a task, as well as the development of that task are carried out by different people (project manager, and development team respectively), and these two actions are evaluated separately. In the case of effort estimation, the success of the action is measured by how close the estimations are to the real results (either above or below). In the case of the development of the tasks, developers are rewarded by how efficient they are in carrying out that task, in terms of development effort. These rules could also take into account the quality of the code (using the *quality* attribute of the *TaskBehaviors*), if code reviews, or testing of particular tasks are carried out by the company. In some cases it may not be possible to absolutely guarantee good behavior (for example, the project manager and the developers could agree to cheat on the estimations for mutual benefit, but this is highly improbable). Nonetheless, the design of the rules of the game must be cautious with these design traps, trying to detect and avoid them.

All these elements, game mechanics, game economy and game rules can be combined in the analysis and design of game cases, that is, in the use cases of the gamified environment.

3.2.5. Development of the gamified SE platform

The purpose of this task is to move the solution designed in the previous tasks to the working platform of the organization. This task is of course completely dependent on some characteristics of the organization, such as the maturity of the processes of the organization and its tool ecosystem. The input for the gamified platform is those actions carried out by the players; these can be performed on the software tools that already exist in the organization. For instance, if the organization wants to gamify a working process that is not currently supported by a tool, it

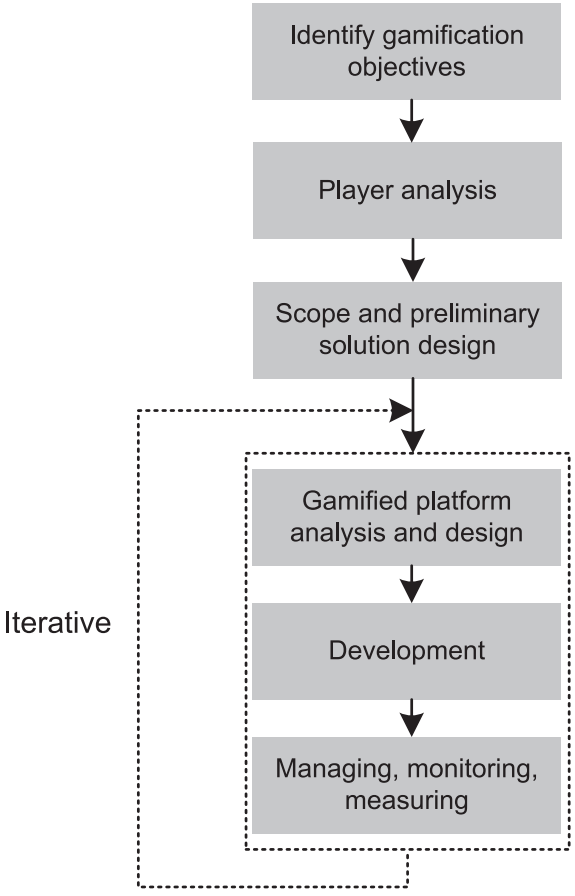

Fig. 6. GOAL: methodological support.

may make the gamification of that process more difficult, and the organization could consider setting up a tool for that purpose. Even if the process is currently supported by a tool, several aspects need to be considered, such as the possibility of modifying the tool, introducing new behavior into it through plugins, or at least accessing the data generated during the use of the tool.

We consider that the best choice for the development of the gamified platform is to follow an agile methodology like Scrum. The book from Keith (Keith, 2010) particularizes this methodology, applying it to the case of game development; (although game development is not the same as the development of a gamified environment, these two cases share some aspects). Actually, not only should this task be iterative, but the whole set of "game analysis and design, development of the platform, management and monitoring" should be iterative (see Fig. 6). This approach would allow the organization to introduce the different elements of the gamified platform incrementally, obtaining early feedback on the design of the platform that would enable decisions made in earlier phases of the process to be corrected or improved.

3.2.6. Managing, monitoring and measuring

The last task of the methodology is devoted to analyzing and evaluating the gamified environment that has been developed and to considering its effect on the work processes it has to do with. This task will allow us to detect potential problems in the functioning of the gamified platform, and to correct past decisions or to consider different alternatives. In the case of those business goals for which it is possible to measure performance through quantitative or qualitative indicators, this task should be based on those indicators.

In addition, the monitoring and measuring task ought to consider other important aspects apart from the achievement of purely business goals, such as the impact of the gamified platform on the players. In other words, we should also evaluate how the players value the gamified environment with respect to their previous work environment in terms of motivation, engagement, fun, and other aspects such as social activity, continuous information of their own actions, and their associated achievements, etc.

The sources of data for this task are therefore multiple. The tools that gather the information of the gamified environment contain data that can be used directly, but in some cases it may be necessary to conduct surveys on the employees. Although originally proposed for the assessment of stakeholder experience in serious games, an assessment methodology like SGSEAM (Xu et al., 2013; Xu, 2015) could help the organizations to carry out this evaluation. An interesting line for future work would be to develop a specialized methodology for the assessment of gamified environments, taking SGSEAM as a basis.

If the organization has a mature quality assurance system, this task can be performed as part of the measurement and analysis processes. Gamification is an additional practice incorporated to the work system of the organization that aims at improving the overall results, so it makes sense to incorporate the monitoring and improvement part of this methodology to the measurement, analysis and continuous improvement processes. If the company does not have mature quality assurance and continuous improvement processes, the organization should define when the monitoring must be done, and who would be the responsible for this task.

As happens in any process improvement effort, this task is limited by the maturity of the measuring and analysis activities of the organization. Those with a mature measuring framework will be able to quantitatively analyze the impact of gamification in their work processes. Those organizations with simpler (or non-existent) measuring systems will have to rely more on surveys of their people. For those organizations with an advanced measuring system, El-Nasr et al. (El-Nasr et al., 2013) provide a framework on how to develop a scorecard based on player data; this can be applied to a gamified environment. Lastly, the results from the monitoring and measuring activities may raise some problems in the functioning of the gamified environment; these should generate new iterations on the process, or at least improvement actions to correct those issues detected.

3.3. GOAL supporting tool

As we explained in Section 2, one of the problems we have identified in existing research in gamification in SE is the difficulty of integrating the gamification proposal into an organization's workplace. To the best of our knowledge, and by considering the updated literature review which was conducted, there is not a research solution to this problem. While generic solutions have appeared for gamification, such as Badgeville[8] for example, we found them focused mainly on domains such as marketing and learning, based on simple game mechanics. We haven't found any generic tool for gamification in software engineering which tackle the particularities of this application domain.

In most cases, previous research work on gamification in SE implementing their gamification proposal through a support tool have developed a specific tool for that purpose. That is, if the authors proposed a way of gamifying requirements management, they have usually developed their own gamified requirements management tool. The same could be said in other software process areas. While it is true that those tools are usually presented

[8] https://badgeville.com/.

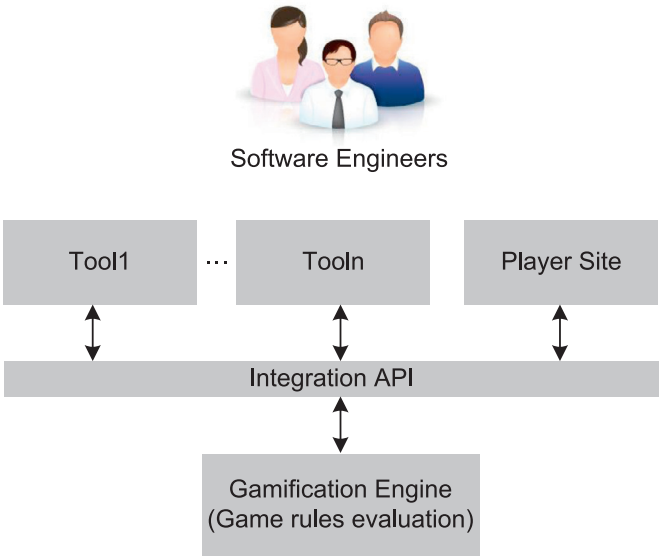

Fig. 7. Gamification Tool architecture.

as a proof of concept, in our opinion it is not clear how those gamification techniques could be moved to the working tools of a real organization. Any minimally-mature software development organization relies on a suite of tools that support the different aspects of a project. Those tools may have been developed ad-hoc by those companies, or they can be existing products (either commercial or open source). In any case, the adoption of those tools is usually the subject of intense debate, effort, and even investment. On the other hand, prototype gamified tools are usually far from existing products at a functional level. Replacing one of the existing tools by a gamified one can thus be very difficult.

Another possibility could be to modify the tools the company is currently working with, to gamify them. While this may seem a better option, it also has its drawbacks. In this paper we consider an integral approach for gamification in software engineering. That is, we do not limit the application of gamification to a particular software process area, but we believe it can spread to as many process areas as necessary. In this last case, the gamification of all tools involved should be integrated in some way.

This led us to the gamification support tool we present in this subsection, which is depicted schematically in Fig. 7. The main component of our proposal is a gamification engine that receives all the behaviors carried out by people in the working tools; it then evaluates them to determine if they deserve a reward, according to the set of gamification rules specified by the designer of the gamified environment. The gamification engine thus stores a log of all the actions completed by each person, the gamification rules, and the rewards corresponding to each action.

The behavior of the gamification engine is simple from a functional point of view. The gamification designer defines the types of action that will be considered in the environment. A gamification rule can be viewed as a function that assigns (or not) a reward to an action of a given type depending on the type of that action, and according to whether that action fulfils the condition of the rule.

Example 1: If the designer wants to reward the completion of a development task, a type of action "Task Completed (TC)" would be defined in the gamification engine corresponding to that behavior. The designer could thus define as many rules as necessary to evaluate those actions. For example, we could define a rule for evaluating TC actions with the condition that, if the action was completed with an effort less than or equal to the effort estimated in the project plan, the player will receive a reward in the form of as many points as the number of hours estimated for completing the task. When the gamification engine receives a TC action from some tool, it automatically triggers the evaluation of the rule, which leads to the assignment of the reward if the action fulfils the condition.

Example 2: Fig. 8 shows a screenshot of the GOAL support tool in which we have created a rule to evaluate the behavior consisting in creating a new task in the project management system. As we can see in this Figure, a single behavior could have more than one associated achievement. In this example, users will receive a badge and two points of experience for each task registered. In addition, the users will receive an extra badge the first time they create a task. Notice that the support tool does not force the users to work with a particular project management tool.

The next point to be solved is how the tools can communicate the actions of the players to the gamification engine. To facilitate this integration, the gamification engine provides an integration API, implemented as a REST web service. This supports the integration in two ways: If the tool to be integrated can be modified or completed through the development of plugins, then the REST service can be used directly in that way; if that option is not possible, we can develop an intermediate connector that mediates between the tool and the gamification engine. The details of the development of that connector depend, of course, on the particular tool to be integrated.

The gamification engine has been used in the application scenario we present in the next section, but also in other settings. Many widely-used tools have been already integrated with it; for example, JIRA, Redmine, or TestLink.

When a tool sends an action to the gamification engine, it receives the result of the evaluation of that action, so it can show the player the reward achieved by means of that action. It is, however, difficult to provide an appropriate interface for a gamified environment in one or several working tools. In the application example we present in the next section, we have completed the architecture with a player website (see Fig. 9). This site is an application focused exclusively on the gamified environment. This means that players can see the log of their actions on this site, along with the rewards obtained from each of those actions, their level in the game, and different rankings comparing their results with that of their workmates, etc. This provides the players with immediate feedback on the results of their actions (continuous feedback is considered by many authors as an important gamification element (Zicherman and Cunningham, 2011; Hugos, 2012)).

In our case, this player site also provides the players with many elements present in pure games, such as social network, messaging, and chatting. Of these elements, the social network may be the most interesting for a gamified environment. For instance, in organizations where the number of players is large an employee does not necessarily know all of his/her workmates. Therefore, the comparison of their results, which belong to people who do not even know each other, can be irrelevant. However, if we have explicit friendship relationships, this allows us to provide the players with more relevant information.

Finally, it is important to remark that the GOAL framework components could be separately applied. For instance, the ontology can serve by itself to characterize the involved concepts about gamification for SE environments and promote a shared understanding about this domain among the involved stakeholders. Furthermore, the ontology and methodology can be applied without relying on the supporting tool we have presented in this section. The gamification engine we have developed has served as a proof of concept of the suitability of the GOAL framework, and to support the case study which is presented in the next section. Other gamification solutions could be applied in combination with

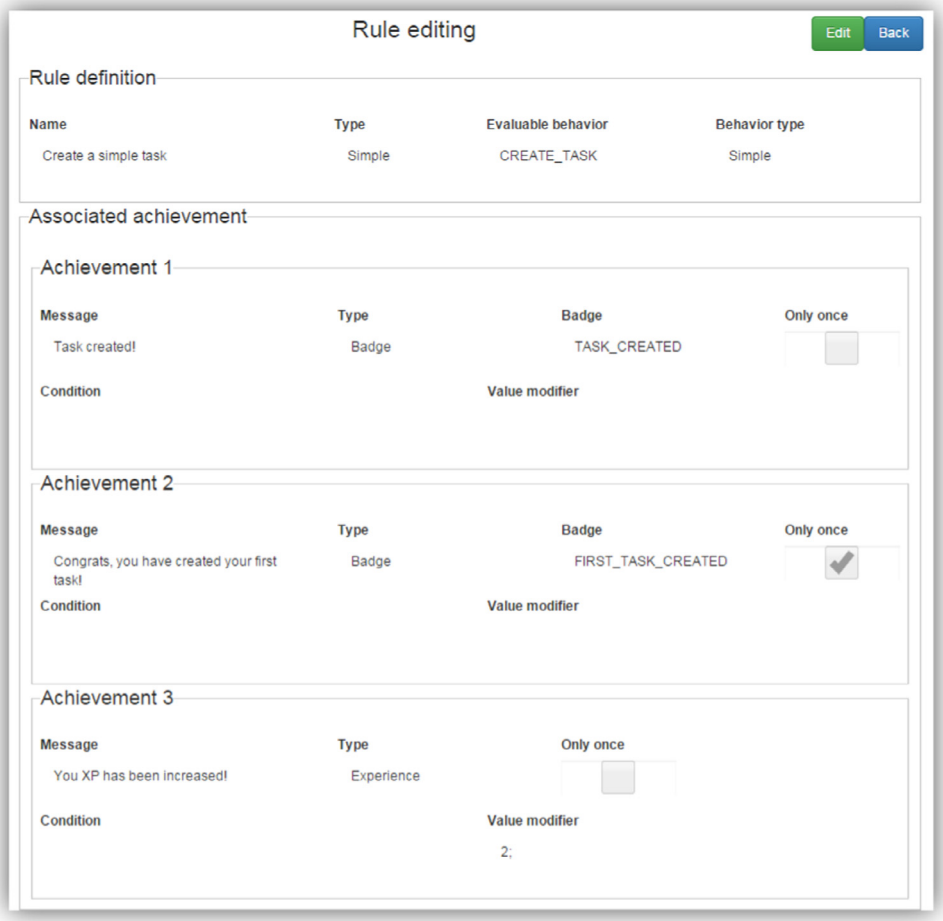

Fig. 8. Example of rule definition in the gamification engine.

the GOAL ontology and methodology, depending on the needs of each company.

4. Application of GOAL in a software company

We have used the case study method to apply the proposed framework in a real software company. The protocol template for case studies presented in (Brereton et al., 2008) and the guidelines proposed in (Runeson, 2012) were followed. The following subsections describe the study in terms of background, design, subjects and analysis units, field procedure and data collection as well as intervention, along with an analysis of results obtained and lessons learned.

4.1. Background

4.1.1. Description of the company

The case study has been carried out in a software development company. In the rest of the paper we will refer to the company as SWComp. SWComp is a Spanish small/medium company focused on software development, services, and IT systems operation. It was founded in 2004 and works at both a nationwide and an international level. SWComp currently employs 25 people, 19 of whom are devoted to software development, as developers, analysts, or project managers. The company has developed software products that are offered as off-the-shelf packages, sometimes slightly adapted to the needs of each customer. SWComp also provides services of custom software development. Its main areas of expertise are geographic information systems, applications for creating and managing digital contents for education, cultural organizations or publishing houses, electronic commerce, and business process support systems. In addition to software development, the company also provides services of software systems operation and IT infrastructure management.

The firm has previously carried out efforts in quality management and improvement. Since 2009, SWComp has implemented and maintained a security management system (SMS) certified under the standard ISO 27001 (ISO/IEC 27001:2005 2005). Regarding software development and software project management, the organization has established and maintained a software project management system based on ISO 15504/ISO 12207 (SPICE) (ISO/IEC 2003; ISO/IEC 2008), since 2011, taking development practices from the Scrum (Schwaber, 2004) agile methodology. This system currently fulfills the requirements of the following ISO 12207:2008 processes: life cycle model management, project planning, project assessment and control, stakeholder requirements definition, software requirements analysis, configuration management, software configuration management, measurement, and software quality assurance. In addition, SWComp has recently taken its first steps in software product quality assessment within the framework provided by ISO 25000 (ISO, 2011).

4.1.2. Tool suite

The tools used by SWComp have evolved with the process improvement efforts of the company. The main support tool for the management of their projects is a custom-developed software

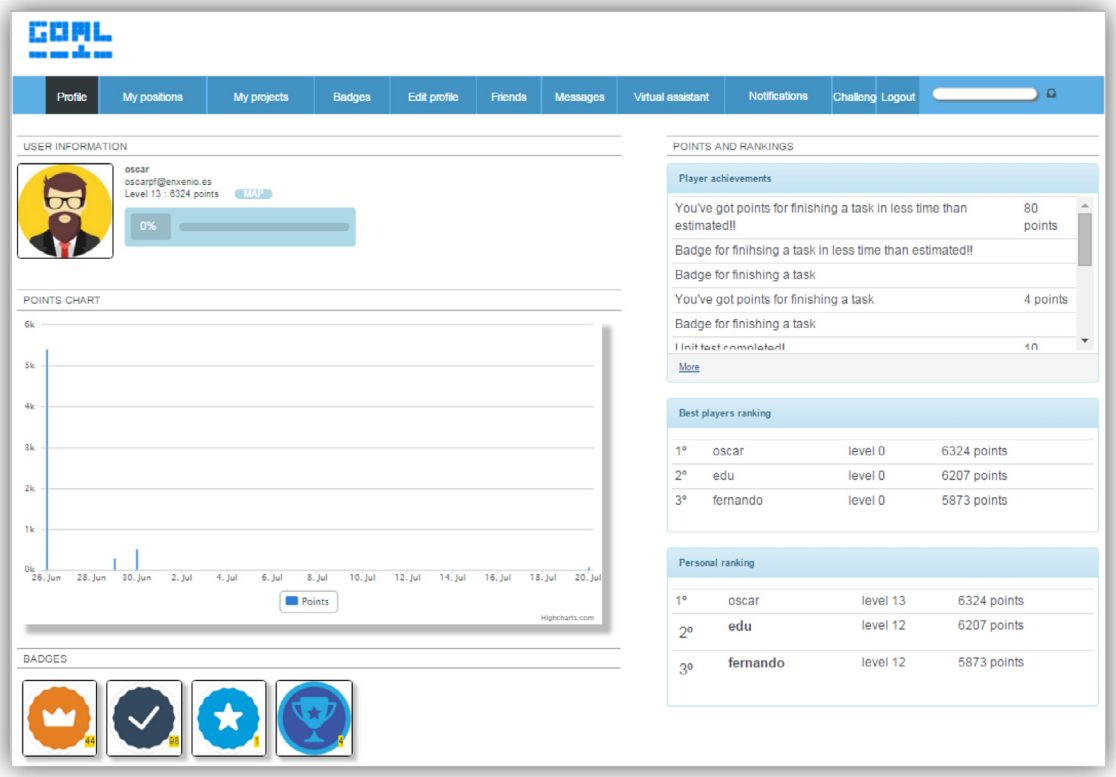

Fig. 9. Screenshot of the main page of the player website.

for project management. This tool allows project managers to create their projects, define the project teams, introduce the tasks composing the projects, with estimates for start and finish dates, and specify the required completion effort for each team member assigned to each task. Each team member therefore accesses the tool on a daily basis to introduce the effort devoted to each task. With that information, the tool provides real-time information on the evolution of the projects, as well as historical reports and forecasts about the predicted project completion figures. Although project planning, management, and monitoring is the primary purpose of this tool, it also provides other functionalities, such as economic and financial management of the project, integration with configuration management systems for baseline management, and internal project audit functionalities.

As part of this project, SWComp has extended this tool to also support requirements management. Each project has an associated requirements book that can be edited by the project manager and whichever team members he/she chooses. The requirements book not only contains descriptions of the requirements of the system, but also allows for collaborative analysis and clarification of the requirements through the comments of the team members, as well as monitoring the state of each requirement (registered, in development, testing, completed). The requirements can be opened (that is, it allows requirements to be added or modified), or closed (no changes are allowed). Every time the requirement is opened and then closed, a new version is saved. This allows version control on the requirements of the projects. SWComp also uses other support tools in its projects, as is the case of Redmine[9] for issue and change management, and JUnit[10] for unit testing.

[9] http://www.redmine.org/.
[10] http://junit.org/.

4.2. Design

Taking the approach presented by Yin (2013), the design type of the case study undertaken in this work is single case – holistic, since GOAL has been applied in the context of one single case in which SWComp applied GOAL in a project for gamifying three main process areas of the organization, namely requirements management, project management, and software testing. The object of study is the GOAL framework we presented in the previous section.

The main research question of the case study was: Is it feasible to apply GOAL to promote the integration of gamification in software engineering environments? The feasibility of GOAL is particularized in the following aspects: results, benefits, and limitations of the application of the GOAL methodology, and suitability of the software architecture and support tool presented. The following table summarizes the main (MRQ) and specific research (SRQ) questions of the case study (Table 2).

4.3. Subjects and analysis units

SWComp was described in the previous sub-section. The company, along with other different firms, participated in the case study as part of a wider research project devoted to the application of gamification in software engineering. In the development of the case study they had the help of the authors of this article. The analysis units in the case study are (i) the GOAL ontology and methodology, and (ii) the support architecture and tool.

4.4. Field procedure and data collection

The field procedure of the case study is directly determined by the activities of the GOAL methodology: identification of business objectives, player analysis, scope and solution definition,

Table 2
Research questions of the case study.

Research questions	
MRQ	Is it feasible to apply GOAL to promote the integration of gamification in software engineering environments?
SRQ1	Is the GOAL methodology suitable for guiding the efforts of a gamification project in an organization?
SRQ2	Is the proposed software architecture and support tool a suitable approach for gamifying a software engineering environment?
SRQ3	Is the effort required by the application of GOAL appropriate with respect to the results and benefits it presents?

game analysis and design, development of the gamified platform, managing, monitoring, measuring.

The data regarding the execution of each of these activities was stored in documents, following the templates of the methodology. Data were also obtained from direct interviews with the team in charge of carrying out the gamification project at SWComp. Additional data, such as efforts of each activity, was obtained from SWComp's project management system.

4.5. Intervention

This subsection summarizes the main aspects of the execution of the case study applying GOAL in SWComp. The SWComp's project team was composed of three people. All of them participated in the design of the gamified environment, and two of them were in charge of connecting SWComp's tools with the gamification engine that centralizes the evaluation of the player's actions. The authors of this paper participated as advisors in the application of the GOAL framework during the case study.

In the rest of this section we describe both the application of the methodology and the resulting gamified software engineering environment of the company.

4.5.1. Business Objectives, player analysis and scope of the gamified environment

The business objectives of SWComp in this project can be grouped around three process areas, namely project management, requirements elicitation and analysis, and test and issue management. These three process areas are usually the most basic ones for any SME software development company. This led the company to establish a total of fourteen business objectives:

- *Business objectives involving project management*: improve estimates, improve productivity, activate project monitoring, complete information of projects.
- *Business objectives involving requirements management*: use of the platform for recording requirements, monitoring of the state of the requirements, change management, effectiveness in requirements management, collaboration in requirements management.
- *Business objectives related to testing and issue management*: test effort reduction, improvement of test effectiveness and efficiency, active issue management.

Each of these business objectives was analyzed in terms of the current and desired scenarios, as well as regarding related indicators, where possible. Table 3 shows an example with the third business objective related to project management, "Active project monitoring":

4.5.2. Player analysis

The player analysis is, along with the identification of the business objectives, the main input to the design of the gamified environment. In the analysis of the players, SWComp used the templates we present in Appendix A, with the Bartle's player taxonomy. The scope of the gamified environment includes all software project management and development activities; so, according to the company's structure, there are two types of players involved in the gamified environment: project managers and team members. In order to carry out the player analysis, SWComp included all the project managers (five), and a sample of ten developers from the company, assuming that the set of developers is a representative sample of the rest of the company. These player types were characterized according to the proposal made by Bartle (Bartle, 1996). The percentages of each category for each player type were established in a subjective way, by categorizing the people in the sample and getting the percentages of each category for each player type. This task was performed by the SWComp team, with the help of the research advisors. The results are shown in Table 4.

In addition, the company considered other demographic and psychographic characteristics in the analysis (age, years of experience, education level, orientation of the people towards technical or managerial work, etc.). SWComp's staff can be characterized as young (the average age of the people in the sample is 29.77), with 4.59 years of experience, and mainly motivated towards technical work.

As part of the player analysis SWComp also considered the organizational culture of the company. Taking into account that the company is only 11 years old, the main characteristics of the organizational culture are:

- *Agile approach*: work culture prioritizing tangible results for the customers over other artifacts derived from the projects. That is, auxiliary artifacts that are not the final deliverables for the customer are kept in their simplest and most agile form.
- *Team qualification*: strong qualification of people involved in development tasks is highly valued. All employees in the role of project managers or team members have a degreein computer science. Most of them also have a Master's degree, and those who do not have one are encouraged to complete it in parallel with their work at the company.
- *Informal work environment*: the organizational structure is characterized by a very simple hierarchy (management, project managers, team members), flexible timetables, informal and close relationships between all the employees.
- *Young organization*: both in terms of the age of the company and the average age of its employees.

Potential intrinsic and extrinsic motivators were also identified as part of the analysis:

- *Intrinsic motivators*: fulfillment of the person's own work-associated duties, contribution to the company's results, professional development, attractiveness of the tasks (although relevant only in some cases, not everybody's).
- *Extrinsic motivators*: public recognition of the results achieved in the tasks, competition with workmates, influence of real rewards in the company (incentives, promotions, etc.).

4.5.3. Solution and gamified environment

The analysis of the company, its business objectives, the organizational culture, and the player types served as the basis for the design of the gamified environment. In this section we outline the scope of the solution, that is, which process areas were gamified and which company's tools were affected, the solution adopted in terms of the game mechanics and economy, and how this design was implemented in the workplace.

Table 3
Example of business objective.

Id	P3
Objective	Active project monitoring
Description	To guarantee that project monitoring and control is done on a frequent basis, through the creation of checkpoints (project snapshots used for productivity and result forecast, and for progress reports).
Current scenario	Project monitoring and control through the creation of checkpoints is not done with the appropriate rigor, yielding information which is poorer than desired, with numerous non-conformities to the organization's procedures.
Desired scenario	Projects are monitored with a frequency that is higher than or equal to the threshold required by the procedures (at least every two weeks).
Indicators	Checkpoint frequency.
Objective value	2 weeks.

Table 4
Characterization of player types according to Bartle's taxonomy.

Player Type	Killer	Achiever	Socializer	Explorer
Project manager	20%	100%	80%	80%
Team member	0%	60%	60%	40%

4.5.3.1. Scope of the gamified environment. From the business objectives identified in the first task of the process, the gamification affected the process areas of requirements management (from elicitation to requirements monitoring), project planning, monitoring, and control, as well as the management of the test efforts and issue management. Both the requirements and the project management are supported by the same tool in the company.

4.5.3.2. Solution: game mechanics and economy. As a result of the analysis of the player types, we concluded that no player type falls completely within one single player prototype. However, the player analysis revealed that most players are achievers, socializers, or explorers. This led us to include the following game mechanics:

- *Points and levels*: the main reward players can obtain from their actions will be in the form of points. The more significant the result achieved by a player in an action, the higher the number of points obtained as a reward.
- *Levels*: As usual, the points will determine the level of each player in the environment. The function that maps the amount of points to levels is exponential; this means that achieving the next level becomes more difficult as the player progresses in the game. This design choice is typical in most existing videogames.
- *Rankings*: will allow some form of competitiveness, by allowing players to compare their results with those obtained by their colleagues.
- *Badges*: will be used as a reward to recognize the achievement of certain milestones as the player progresses in the environment.
- *Social interaction*: the player analysis revealed socialization to be an important aspect for players in the company (which is to be expected in such a small organization). Consequently, social interaction will be supported in the gamified environment. The player site will also act as a social network, in which players can have friends with whom they can chat or send messages. In addition, this social network also affects other game mechanics such as rankings, so players will see a ranking comparing them with the rest of players in the company, as well as a ranking comparing them with their friends.
- *Continuous feedback*: the players will have a player site providing them with information about the results of their actions in real time.
- *Challenges*: this game mechanic was not introduced into the environment for any particular task, but as a way to promote competitiveness. Players can challenge their friends to obtain a given number of points in a given period, for example.

The reason for this choice is that we believe achiever players will find points, levels, badges, and rankings to be game mechanics that allow them to see their achievements in their work continuously, giving them the chance to compare them with those achievements of their friends. Social interaction (in social networks with friendship relationships, and in chatting) will allow the socializer-type players to interact with their workmates, and even compare their achievements to those of their friends. Finally, challenges will allow the players to come up with their own personalized game challenges, giving them something else to add to the fixed rules of the gamified environment. Finally, the inclusion of continuous feedback was not motivated by any particular player type, but is usually considered as fundamental in any gamified environment. Since the number of players under the *Killer* category is quite small, SWComp decided not to include any game mechanic oriented to that type of player such as, for example, direct competition between players in which only one of them wins and obtains a reward.

The design of the rules of the game was determined by the business objectives, the player types, and the game mechanics we have decided to include. The design of these rules was carried out with the help of the template provided in Appendix A.3. Table 5 shows an example of a very simple game rule that rewards the registration of a new requirement in the platform with a fixed number of points.

Table 6 shows how the rule for rewarding completed tasks has been implemented in the gamified environment. As we can see in the definition of the rule, the player will always receive a badge when he/she completes a task. If that task has been completed in less time than estimated, the player will receive an additional badge, and a number of points equal to the number of estimated hours for the task. If the task has been completed with more effort than that estimated, the player will receive a number of points equal to the estimated effort, minus a penalization equal to the number of additional hours he/she needed.

4.5.4. Implementation and deployment

Fig. 10 summarizes the software architecture of the gamified environment. As explained in the previous section, our framework for the gamification of SE environments proposes to centralize all the gamification rules and the evaluation of the player's actions in a centralized game engine. SWComp has used that gamification engine in the gamification of its tools.

As we can see in the figure, EnxManage, the tool used by SWComp to manage projects and its requirements, is integrated with the gamification engine through an integration API. The tool communicates player's actions to the gamification engine through this channel. The integration with other tools used by SWComp will be done through EnxManage too. The reason is that this tool already accesses information hosted in TestLink and Redmine, so

Table 5
Example of game rule: create a new requirement.

Game rule	CREATE_REQUIREMENT
Name	Add a requirement to the project
Description	The user adds a new requirement to the requirements book of the project
Related business objectives	Use of the platform for managing requirements
Target players	All team members
Related game mechanics	Points
Preconditions	None
Evaluable behavior	*CreateRequirement*
Scenario #1	*Grant 10 points*

Table 6
Example of game rule: complete a task.

Game rule	COMPLETE_TASK
Name	Complete a task in EnxManage
Description	This rule is activated when a team member completes a task in EnxManage, and rewards the task in terms of the real effort compared to the estimated effort.
Related business objectives	Improve productivity
Target players	All team members
Related game mechanics	Points
Preconditions	None
Evaluable behavior	*TaskCompleted* (a *TaskBehavior* action)
Scenario #1	*Grant badge "Task completed"*
Scenario #1	*If (realHours<estimatedHours) Then points=estimatedHours;*
Scenario #2	*If (realHours<estimatedHours) Then grant badge "Task completed in time"*
Scenario #3	*If (realHours>= estimatedHours) Then points=estimatedHours - (realHours - estimatedHours)*

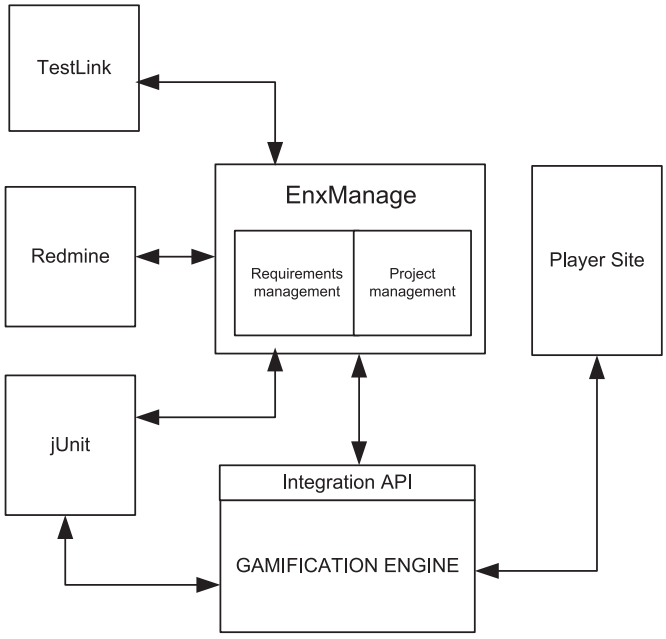

Fig. 10. Software architecture of the gamified environment.

it is easy to use those integrations to extract information about a player's actions and to communicate them to the gamification engine.

SWComp has also developed a player site that shows all the information of the gamified environment. Players access this site as a social network in which all the information (actions, rewards, social network, messaging, rankings, challenges, etc.) is shown.

4.6. Analysis of results and lessons learned from the case study

In this section we analyze the results obtained, the assessment of the GOAL framework, and lessons learned from the case study. The assessment of the GOAL framework is carried out following the three specific research questions we formulated in the design of the case study, and focusing on the three components of the GOAL framework: ontology, methodology, and support tool.

4.6.1. Ontology

The project team of SWComp considered that the ontology provides a complete metamodel of gamification of software engineering environments that reflects the scenario of most software companies and which should therefore fit most of them. The metamodel served as a complete basis for the design of the gamified environment, therefore making the project much easier, since the design of the gamified environment (at least in part) was a specialization of the metamodel provided by the ontology.

In addition, the ontology provides a common set of concepts about gamification, not easily obtainable from existing sources in the field. The field of gamification is quite new, so some authors consider concepts that may be ignored by others, and sometimes use different vocabulary to refer to the same element. Moreover, the ontology links the concepts of gamification (developed in fields such as marketing, education, etc.) to concepts of software engineering environments, therefore specializing the field of gamification for these last scenarios.

A potential limitation of the proposed ontology is that it could be incomplete; that is, it may have missed some concept, or it could become incomplete as the field of gamification evolves. Despite this potential limitation, it provides a sound basis that could incorporate new concepts in the future.

4.6.2. Methodology

Existing literature on gamification presents many aspects that should be considered when gamifying an activity or environment, but most of them present those aspects without any kind of methodological support. Even when something like a methodology is outlined, it is usually very general and without any further support.

The methodology is the component of the GOAL framework that has been most valued by the team in charge of the gamifica-

tion project at SWComp. This team highlighted the following main benefits from the methodology:

Guided process: the methodology provides a step by step, iterative approach for gamification, supported with templates (a selection of the most relevant ones presented in Appendix A), examples (included in the description of the methodology, or even in the templates), and external sources for many activities (such as Bartle's or Marczewski's taxonomies for player analysis). This made the process very direct, as we always knew what the next step would be, and most importantly, we knew exactly what to do in each activity. The templates and guidelines associated with each activity made its execution simpler, and the quality of the results much easier to obtain. From the different activities considered in the methodology, the first two were the ones most valued by SWComp's project team, since their outputs determined to a high degree the game mechanics and game rules defined in the following activity, the design of the gamified environment.

- *Alignment with the business objectives*: the team highlighted the benefits of the first task of the methodology. Considering this aspect in a systematic way allowed the team to relate this project's goals to the goals of the company and its existing working processes, allowing them to identify areas for gamification that would not have been considered otherwise. In addition, this allowed the company to link some of those objectives and their related gamification elements to the metrics of their measurement and analysis processes. During the design of the gamified environment, the list of the business objectives determined the list of the game rules.
- *Player analysis*: this activity, together with the links and guides on how to apply existing player analysis proposals, turned out to be very valuable to the team. The project team concluded that the analysis of the player types, together with links to appropriate game mechanics associated with each of them, was very valuable, since the general solution and most game elements incorporated into the workplace were directly implied by this player analysis. As we have seen in previous sections, the game mechanics included in the gamified environment were selected in terms of the most usual player types in the organization.

Therefore, from the analysis of the results obtained in the case study, and since SWComp's team valued all the activities of the methodology positively, we concluded that it is feasible to use the GOAL methodology to guide the efforts of a software organization towards incorporating gamification into their working environment (SRQ1).

4.6.3. Support architecture and tool

This component of the framework played a very important role in the project. As we explained in the presentation of related work (Section 2), most existing works on gamification in SE provide custom gamified tools as the way of incorporating gamification into an organization. However, the SWComp team rejected that possibility. The use of GOAL's software architecture and gamification engine allowed SWComp to gamify the tools they were already using without having to replace any of them or introducing any change in their work processes. If this approach had not been followed, the scope of the gamification project would necessarily have been narrower, since integrating behaviors from such different tools would have turned out to be complex and costly. SWComp was able to integrate with little effort, as we will see next, very different tools, such a custom-developed project management software, and off-the-shelf applications such as Redmine, TestLink, or JUnit.

We thereby conclude that the proposed software architecture and support tool clearly helped the organization to adopt an integral gamification solution adapted to their tools (SRQ2).

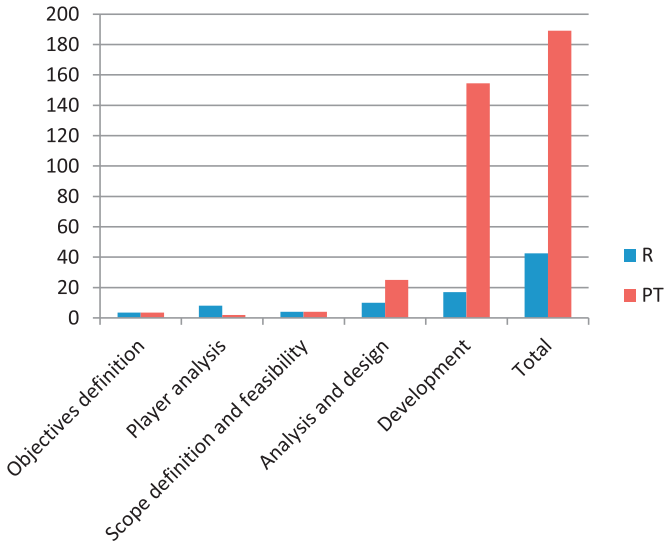

Fig. 11. Summary of efforts required by the gamification project.

4.6.4. Required effort

Fig. 11 shows a summary of the efforts required by the gamification project in SWComp. This figure distinguishes between the efforts of two roles: a researcher (R), and the project team (PT). Basically, the researcher gave support to the project team in the understanding and application of the GOAL framework in SWComp. As we can see in the figure, the total effort from the researcher is 42.5 hours, while the effort of the project team is 189 hours. The sum of both is 231.5 hours. According to the person responsible for the project in SWComp, this effort was considered appropriate, and would have been greater without the guidance of the GOAL framework. This being so, we conclude that the application of the gamification framework can be done with a reasonable effort (SRQ3).

4.6.5. Gamification benefits

The benefits of applying gamification practices in SE workplaces and educational environments have been proved in much previous research work (see Section 2). Of course, the quality of the gamified environment will affect those benefits. Although this aspect was initially out of the scope of the case study (to allow there to be a sound statistical study results and quantitative data on a real setting for a long and significant period of time are required), the employees of SWComp positively the introduction of these mechanics and practices into the workplace valued. They highlighted, for example, that it adds a "fun" component to their daily work, that it allows them to see their progress explicitly and compare it with that of their workmates, and even that it is a valuable source of information for the company's management. This is because the gamified environment provides a complete log of the actions of each person, along with the reward the company assigns to them.

While defining the scope of the case study, the possibility of conducting an exhaustive quantitative analysis on the improvement in different business indicators due to the incorporation of gamification to the work environment of the company was considered, but it would not have been possible to isolate other possible confounding factors, such as the usage of new tools in the company, modifications in existing tools, training for the employees, etc. In this sense, it would have been a great challenge to objectively separate the effects due to gamification from the effects due to other actions implemented at the same time. Anyway, the comments and analysis of this case in SWComp allowed us to obtain very useful insight regarding the performance

improvement caused by the gamification, namely: For example, SWComp reported that the collaborative use of EnxManage for requirement analysis improved by more than a 20% (this number is approximated and provided by the company), that is, the number of registered requirements and comments grew by more than a 20%, compared to projects of a similar size completed before the incorporation of gamification. Similar results were obtained for issue management, that also experienced a growth in the number of issues and comments on those issued registered in the Redmine of the projects (again, compared with projects of a similar size completed before the case study was conducted). Although, as a result, some partial conclusions can be extracted from this case study, the potential benefits of incorporating gamification to SE have already been demonstrated by many previous works on the field (as we explain in Section 2) and this study gives some insights which confirm such advantages and can help to guide organizations to obtain such benefits.

A sounder statistical analysis of the measures associated with the business objectives remains as future work, since it requires a gathering of data on those measures over a long and significant period of time, or the conducting of controlled experiments with control groups. We believe that the following aspects should be considered for such an experimental study:

- In order to obtain accurate quantitative results, gamification should not interfere with other improvement actions being implemented at the same time. This may be difficult to achieve, but using the gamified environment in a selection of pilot projects would be a way of getting to it.
- We believe that a study like that should span over a long period of time, since one of our concerns is how the effects of gamification in the workplace evolve over time.
- Finally, it would be desirable that many studies with this goal were conducted in organizations with different profiles, to avoid results biased by the particular organizational culture of a given company.

4.7. Validity threats and limitations of the case study

The design of the case study was carried out according to recognized best practices and advice on the subject (Runeson, 2012). In order to address potential threats to the validity of the case study, we have made the following considerations:

- Construct validity: before starting the execution of the case study, the researchers held meetings with the project team at SWComp to explain the goals and research questions of the case study to them, and to clarify the components of the GOAL framework, as well as the way the research questions would be evaluated. The results of the interviews with the project team were revised with them, in an effort to avoid potential misunderstandings.
- Internal validity: in order to avoid other factors affecting the result of the case study, the project team was provided with detailed guides on the application of each activity of the methodology, and with links to relevant sources on the field of gamification.
- External validity: some results of the application of the GOAL methodology depend on the company, significantly, such as the business objectives and the player analysis. The results obtained from applying the GOAL methodology in other companies may differ for this reason. However, most software development companies share the same problems as SWComp, and we believe the results we have obtained are therefore applicable and that they can be generalized.
- Reliability: the gamification methodology we have proposed has associated templates and guides to each of its activities (the templates are not included in the article, for reasons of space). Hence, the application of the GOAL framework in other studies should lead to similar results, as they do not depend on the particular researchers that carried out this study. The nature and characteristics of other companies could be the source for differences in relation to the results we have obtained.

5. Conclusions and future work

The application of gamification in software engineering has gained attention from both the research community and the industry. Although many research papers have been published on the benefits of applying gamification in SE, previous work has identified an important lack of methodological and technical support in the application of gamification. In this paper we have presented a framework for the application of gamification in SE environments. This framework is composed of three elements:

- Ontology: the ontology provides a common set of concepts for gamification in SE environments, and how they are related to each other.
- Gamification methodology: the gamification methodology provides a step-by-step process for gamifying a SE environment, taking the business objectives and the result of player analysis as the base for the design and development of the gamified environment.
- Support tool: we present a support tool that allows organizations to gamify their tools easily. This tool centralizes the definition of behaviors, achievements, and the rules of the game, in such a way that the current tools used in the organization only have to communicate the user's behaviors.

The GOAL framework for gamification of SE environments has been evaluated by means of its application in a real software development company through a case study. The company has applied the proposed framework to carry out an integral gamification of its workplace, focusing on the areas of project management, requirements management, and testing. From the case study, we have extracted the following conclusions:

- The ontology served to develop a complete metamodel of gamification in SE environments, and has served as a complete basis for the design of the gamified environment.
- The methodology has helped the company to integrate gamification practices in its work environment appropriately, by providing a clear and guided path with application guides, templates, and links to helpful sources. The company highlighted two elements of the methodology particularly, namely the alignment with the business objectives, and player analysis as the starting point of the process. The analysis of the business objectives that it is intended will be improved with the project ensures that everything done was aligned to these objectives. The analysis of the players made it easier to decide which game mechanics to incorporate in the gamified environment, as existing work on player analysis has already made suggestions in this line.
- Technological support: previous research papers on gamification in SE usually presented gamified tools as a proof of concept of their approach. However, that kind of approach is not feasible in a real organization. The company found it very easy to gamify its current tools by just integrating them with a gamification engine that centralizes the definition of the rules of the game, and their evaluation, under the communication of a behavior of some player.
- The application of the framework led to a rich solution that goes beyond what is usually referred to as "pointification", since the gamified environment we obtained includes a wide

range of game elements, such as points, levels, badges, quests, dashboards, rankings, visual metaphors, and social interaction and reputation.

The GOAL framework provides the designers a complete solution for applying gamification in SE in a complete way, that is, considering all the phases of the software life cycle, and all the processes involved in each of them: proposal, requirement analysis, design, development, testing, release, and maintenance. The designer would only need to determine the process areas subject to be gamified, analyze the improvement objectives in each of them and the players involved, and then include those process areas in the design of the global gamification environment.

The GOAL framework can be applied not only to development processes, but also to the operations area (deployments, system operation, issue solving, etc.) of software organizations. This can be specially interesting for those organizations embracing the DevOps approach for the development-operations integration.

Future work includes further analysis of the results provided by the gamified environment, and potential adjustments of its design, and new case studies focused on additional quantitative assessments of the benefits during long periods of time. Aspects such as performance, productivity, motivation, and engagement of the players can be considered. The application of this methodology to other domains such as educational SE environments is another interesting line for future work.

Acknowledgments

This work has been funded by the following projects: GOAL (CDTI EXP 00064563 / ITC-20133062), VoxPopuli (Ministerio de Economía e Competitividade) (PGE e Fondos FEDER) ref. TIN2013-47090-C3-3-P), 4V (Ministerio de Economía e Competitividade) (PGE e Fondos FEDER) ref. TIN2013-46238-C4-3-R), MOV-RDF (Ministerio de Economía e Competitividade) (PGE e Fondos FEDER, ref. TIN2015-69951-R), and GRC (Xunta de Galicia, ref. GRC2013/053 (FEDER)) for Oscar Pedreira, Ana Cerdeira-Pena, and Miguel Penabad, and GINSENG (Ministerio de Economía e Competitividade y Fondo Europeo de Desarrollo Regional FEDER, TIN2015-70259-C2-1-R) for Félix García and Mario Piattini.

Appendix A. GOAL Methodology Templates

This appendix presents some of the templates included in the GOAL methodology as a support to its practical application. Although the methodology could be applied without using them (or without using them exactly as we propose them), we believe they can be helpful in most cases. Not all the templates are shown, for reasons of space, so we present the most significant ones.

A.1. Business Objectives

Table 7

Table 7
Template for documenting business objectives.

Id	*Business objective identifier*
Objective	*Name*
Description	*Description.*
Current scenario	*Description of the current scenario regarding this particular business objective.*
Desired scenario	*Improvements the company would like to achieve.*
Indicators	*Quantitative/qualitative indicators the company could use to analyze the improvements regarding the objective.*
Objective value	*Objective value associated with the indicators identified for this business objective.*

Table 8
Player analysis table.

Player analysis table
Surname, Name
Demographic profile
…
…
Psychographic profile
…
…
Bartle category(ies)

Table 9
Player analysis summary.

Gamified environment		
Organization's culture		
Intrinsic motivators		
Extrinsic motivators		
Player types	**%**	**Additional comments**
Killer		
Achiever		
Socializer		
Explorer		

A.2. Player Analysis

The following table shows a template for the individual analysis of players. The demographic profile could include characteristics such as age, gender, education level, job position, work experience, time in the company, etc. The psychographic profile could include characteristics such as interests, lifestyle, opinions, values related to his/her work at the company, expectations, etc. This template assumes that the players would be categorized according to Bartle's taxonomy (Bartle, 1996). The template could be easily adapted to other player analysis proposals, such as Marczewski's proposal (Marczewski, 2014) (Table 8).

The following table makes it possible to present a summary of the results of the player analysis, summarizing the organization's culture (which can contain data from the player's profiles), and the result of the player's categorization according to Bartle's taxonomy (again, the template could be easily adapted to other player analysis frameworks). This template can be extended with other features, depending on which demographic or psychographic characteristics have been considered in the previous one (Table 9).

A.3. Game Design

The following template can be used in the design of the rules of the gamified environment (Table 10).

Table 10
Game rules definition.

Game rule	*Identifier*
Name	*Name*
Description	*Brief description*
Related business objectives	*From those identified in the first activity*
Target players	*E.g., project managers, developers, testers, etc.*
Related game mechanics	*E.g., points, badges, levels, etc.*
Preconditions	*Preconditions for rule evaluation*
Evaluable behavior	*Type of behavior that will trigger the evaluation of this rule*
Scenario #1	*First evaluation scenario. E.g., If* $\langle \text{condition}_1 \rangle$, *then* $<reward_1>$
Scenario #2	*If* $\langle condition_2 \rangle$, *then* $<reward_2>$

References

"CMMi for Development, Version 1.3," Software Engineering Institute, http://cmmiinstitute.com/resources/cmmi-development-version-132010.

ISO/IEC 15504-2:2003, 2003. Information Technology - Process Assessment – Part 2: Performing an Assessment.

ISO/IEC 12207:2008, 2008. Systems and Software Engineering - Software Life Cycle Processes.

ISO/IEC 25010:2011, 2011. Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models.

DeMarco, T., Lister, T., 2013. Peopleware: Productive projects and Teams, 3rd Edition ed Addison-Wesley.

Curtis, B., Miller, S.A., Hefley, W.E., 2001. People Capability Maturity Model (P-CMM) Version 2.0. Software Engineering Institute.

Humphrey, W.S., 2000. The Personal Software Process (PSP). Software Engineering Institute.

Humphrey, W.S., Chick, T.A., Nichols, W., Pomeroy-Huff, M., 2010. Team Software Process (TSP) Body of Knowledge (BOK).

Glass, R.L., 2006. Software Creativity 2.0. Editorial developer * Books.

Passos, E.B., Medeiros, D.B., Neto, P.A.S., Clua, E.W.G., 2011. Turning Real-World Software Development into a Game. In: Proceedings of the Brazilian Symposium on Games and Digital Entertainment (SBGAMES'11), pp. 260–269.

Deterding, S., Dixon, D., Khaled, R., Nacke, L., 2011. From game design elements to gamefulness: defining gamification. In: Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments (MindTrek'11). ACM, pp. 9–16.

Zicherman, G., Cunningham, C., 2011. Gamification By Design. O'Reilly.

Werbach, K., Hunter, D., 2012. For the Win: How Game Thinking Can Revolutionize Your Business. Wharton Digital Press.

McMillan, D., 2011. Gamification': a Growing Business to Invigorate Stale Websites. Bloomberg, Businessweek.

[Authors-ommitted], 2015. Gamification in software engineering – A systematic mapping. Inf. Software Technol. 57, 157–168.

Hugos, M., 2012. Enterprise Games. O'Reilly.

Hamari, J., Koivisto, J., Sarsa, H., 2014. Does gamification work? - A literature review of empirical studies on gamification. In: presented at the Proceedings of the 47th Hawaii International Conference on System Sciences (HICSS-47).

Xu, Y., 2012. "Literature Review on Web Application Gamification and Analytics," Collaborative Software Development Lab (CSDL). Department of Information and Computer Sciences, 11-05. University of Hawai'i Available http://csdl.ics.hawaii.edu/techreports/11-05/11-05.pdf .

R. Bartle, "Hearts, Clubs, Diamonds, Spades: Players Who suit MUDs", J. MUD Res., 1(1), 19, 1996.

El-Nasr, M.S., Anders Drachen, Flynt, J.P., Salem, O., 2005. Software Engineering for Game Developers (Software Engineering Series). Thomson.

Kumar, J., Herger, M., Deterding, S., Schnaars, S., Landes, M., Webb, E., 2013. Gamification @ work. In: Proceedings of the Conference on Human Factors in Computing Systems (CHI'13). ACM Press, pp. 2427–2432.

Kumar, J.M., Herger, M., 2013. Gamification at Work: Designing Engaging Business Software.

Keith, C., 2010. Agile Game Devlelopment With Scrum. Addison Wesley.

Marczewski, A., 2014. User types & player types in gamification. presented at the Gamification World Congress (GWC'14).

Burke, B., 2014. Gamify: How Gamification Motivates People to Do Extraordinary Things. Bibliomotion.

Dignan, A., 2011. Game frame: Using Games As a Strategy For Success. Free Press.

Jacobson, I., Spence, I., Bittner, K., 2011. Use-Case 2.0 - The guide to succeding with use cases. Ivar Jacobson International.

El-Nasr, M.S., Drachen, A., Canossa, A., 2013. Game Analytics: Maximizing the Value of Player Data. Springer.

Xu, Y., Johnson, P.M., Moore, C.A., Brewer, R.S., Takayama, J., 2013. SGSEAM: assessing serious game frameworks from a stakeholder experience perspective. presented at the Procs. of the First International Conference on Gameful Design, Research, and Applications.

Xu, Y., 2015. Makahiki and SGSEAM: Design and Evaluation Of A Serious Game Framework For Sustainability and Stakeholder Experience Assessment Method. University OF Hawai'i at Manoa.

Brereton, P., Kitchenham, B.A., Budgen, D., 2008. Using a protocol template for case study planning. In: presented at the Proceedings of the International Conference on Evaluation and Assessment in Software Engineering (EASE'08).

Runeson, P., 2012. Case Study Research in Software Engineering: Guidelines and Examples. John Wiley & Sons.

ISO/IEC 27001:2005: Information technology - Security techniques - Information security management systems - Requirements, 2005.

Schwaber, K., 2004. Agile Project Management with Scrum. (Developer Best Practices). Microsoft Press.

Yin, R.K., 2013. Case Study Research: Design and Methods, 5th ed SAGE Publications.

Félix García received his MSc (2001) and PhD (2004) degrees in Computer Science from the University of Castilla-LaMancha (UCLM). He is currently an associate professor in the Department of Information Technologies and Systems at the UCLM. He is member of the Alarcos Research Group and his research interests include software processes, software measurement, agile methods and business process management. Project Management Professional (PMP)® (Id. 1,821,054). http://alarcos.esi.uclm.es/per/fgarcia/

Oscar Pedreira has MSc. and PhD. degrees in computer science from University of A Coruña. He is an assistant professor since 2008 at the same institution. His research interests include data management and engineering, information retrieval, information systems, and software engineering

Mario Piattini is a full professor at the UCLM. He holds the PhD degree in Computer Science from the Technical University of Madrid and leads the Alarcos Research Group. He is CISA, CISM, CGEIT, and CRISC by ISACA. His research interests include software quality, metrics and maintenance. He is the Director of the Joint UCLM-Indra Software Research and Development Center and the Institute of Information Systems and Technologies.

Ana Cerdeira-Pena received her PhD in Computer Science in 2013. Her research interests include: compressed data structures for information retrieval, digital libraries, and mathematical modeling and algorithms design for operational research problems. Currently, she holds a position of Associate Professor in the University of A Coruña (Spain).

Miguel R. Penabad obained his Master (1994) and Ph.D (2001) in Computer Science (supported by Xunta de Galicia grants) at Universidade de A Coruña. He is a professor in the same university since 2000. His main research interests are database query optimization, and algorithms and data structures for information retrieval.