2016 IEEE 24th International Requirements Engineering Conference

Acceptance Requirements and their Gamification Solutions

Luca Piras, Paolo Giorgini and John Mylopoulos
Department of Information Engineering and Computer Science
University of Trento
Trento, Italy 38123
Email: {luca.piras, paolo.giorgini, john.mylopoulos}@unitn.it

***Abstract*—We live in the days of social software where social interactions, from simple notifications to complex business processes, are supported by software platforms such as Facebook and Twitter. But for any social software to be successful, it must be *used* by a sizeable portion of its intended user community. Usage requirements are usually referred to as *Acceptance Requirements* and they have been studied in the literature both for general technology as well as software. Operationalization techniques for such requirements often consist of making a game out of software usage where users are rewarded/penalized depending on the degree of their participation. The game may be competitive or non-competitive, depending on the anticipated personality traits of intended users. Making a game out of usage is often referred to as *Gamification*, and gamification has attracted huge attention in the literature for the past few years because it offers a novel approach to software technology usage.**

This paper proposes a generic framework for designing gamified solutions for acceptance requirements. The framework consists of a generic acceptance goal model that characterizes the problem space by capturing possible refinements for acceptance requirements, and a generic gamification model that captures possible gamified operationalizations of acceptance requirements. These models have been extracted from the literature and they are highly dependent on context (cognitive and social) elements of the intended user community. The proposed framework is illustrated with the Meeting Scheduler exemplar.

***Index Terms*—Acceptance requirements, gamification, context models, goal models, requirements engineering.**

I. Introduction

Nowadays, digital social interactions are central elements in people's lives. Social software, e.g., Facebook and Twitter, are based on this concept and support it massively by offering a wide range of social mechanisms, starting from simple notifications to complex business processes. The key of success, for each social platform, is to maximize the *usage* of system social functions by the majority of its intended users. Therefore, especially for these kinds of systems, Usage requirements are an important aspect to take in account and to define accurately. Usage requirements are usually referred to as *User Acceptance Requirements* (or simply *Acceptance Requirements*) and they constitute a class of often forgotten requirements.

Acceptance requirements have been treated in a more general context as *technology* acceptance, with many applications reported in the literature. For example, a Mobile Remote Presence System [1] for the elderly, a customizing system for the masses [2], supporting project management in order to minimize risks [3], Software Quality Management [4]. Moreover, there are acceptance models in the literature – adopted from Sociology, Psychology and IT – that have been proven useful in determining acceptance needs in different contexts.

Operationalization techniques for acceptance requirements can reside on designing a game, composed of challenges, rewards, penalizations and other different kinds of game mechanics, where the objective is to increase the quantity and improve the quality of user's activities, by engaging the user, as a player, in using the system and participating more actively to it. Moreover, on the basis of typologies of users/players, different game mechanisms and game characteristics (e.g., competitive or non–competitive) can be selected [5]–[8]. This kind of operationalization is referred to as *Gamification*, and, in the last few years, it has attracted huge attention in the literature, because it offers a novel approach to enhance and increase software usage.

We are interested in developing a generic framework for modelling, analyzing and fulfilling acceptance requirements for software systems through gamification. Our objective is to support the systematic design of social software that meets acceptance requirements. We propose to meet our objective by conducting a wide review of the literature to select the most important, effective and representative user acceptance models [9]–[13]. We have integrated elements of existing models to create an *Acceptance Model* based on goal modeling techniques [14], [15]. This model gives a generic characterization of the problem space for acceptance requirements. We have also developed a *Gamification Model* that defines a design space for gamified solutions to acceptance requirements, also through a literature review. This model includes gamification concepts such as point systems (i.e., experience, redeemable, skill, karma, reputation and training points), badges, leader–boards, levels, paths, gamified training (i.e., suggestions, tricks, tours, tutorials, training paths), gamified market (i.e., rewards and market policies of redeeming, making gifts, purchasing), game roles, powers, unlockable powers [6], [7], [16], etc., and the alternative choices a designer has when designing a gamified solution. Our framework, named Agon[1],

[1]Agon (in Greek Αγων) means "game" or "competition", as in Olympic Games (Ολυμπιακοι Αγωνες)

2332-6441/16 $31.00 © 2016 IEEE
DOI 10.1109/RE.2016.43

RE 2016, Beijing, China
RE@Next! Paper

recognizes the importance of understanding game mechanics and dynamics by applying well–known gamification patterns and guidelines [6], [7] in producing an effective gamified design.

Succeeding sections of this status report are organized as follows. In section II, we present the main elements of Agon and in section III we provide further details and examples concerning the models of our framework. In section IV, we illustrate how our framework can be used, by exploiting the well–known Meeting Scheduler exemplar adapted to a Doodle like solution. In section V, we outline forthcoming activities of this ongoing work. Finally, in section VI, we conclude and discuss future work that goes beyond our study.

II. Acceptance Framework Proposal

Agon (Fig. 1) is composed of an *Acceptance Model (AM)*, a *Tactical Model (TM)*, a *Gamification Model (GM)* and a *User Context Model (UCM)*.

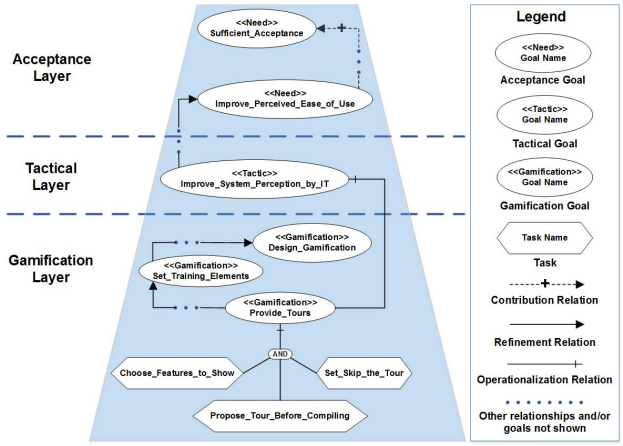

Fig. 1. Abstraction layers of the framework with a simple example.

AM captures psychological factors that can positively contribute towards acceptance of a software system. GM is composed of gamification concepts and best practices, guidelines concerning the gamified design of social systems for maximizing the social software usage. TM is a mediator layer including tactics for refining the problem space of AM and relating its elements to the design space of GM. UCM is the model that describes acceptance–related traits of the users we want to engage.

Moreover, we have enriched our models with *Context Dependant Rules (CDRs)* to describe the circumstances that lead people to use a system. In fact, usage depends largely on *Context*: user characteristics (e.g., gender, age, player type, etc.) and the kind of relation that the users have regarding the Acceptance Subject (e.g., awareness and expertise of the system to be accepted). The literature has many examples of CDRs for improving user involvement in different situations [6], [9], [17], [18]. We model context by extracting from CDRs important dimensions of UCM. Furthermore, we annotate the AM and GM with CDRs to guide the design towards suitable solutions for a given set of acceptance requirements and intended user community.

We designed AM, TM and GM by using and extending the NFR framework [14], [15]. UCM is founded on Context Dimension Trees [19].

AM is a meta–model of stakeholder acceptance *Needs*, with `Sufficient Acceptance` as root–level goal (Fig. 1). Sufficient Acceptance can receive a positive contribution by `Improve Perceived Ease of Use`, and possibly other sub–goals. This goal can be refined by the `Improve System Perception via IT` tactic of TM (goals at the tactical layer are called *Tactics*). In turn, tactics can be operationalized by elements of a gamification solution such as the `Provide Tours` gamification goal. A tour [6] is a demo of the main features of a system where the user is a passive watcher. In the GM, goals are eventually operationalized by *Tasks* [14], [15]. Tasks represent concrete activities that fulfil goals and can be carried out by a person or the system–to–be. For our example, `Provide Tours` can be operationalized with 3 tasks: `Choose Features to Show`, `Propose Tour Before Compiling` and `Set Skip the Tour`. This option is further discussed in section IV.

Agon has different abstraction layers, as shown in Fig. 1. The acceptance layer identifies psychological needs that lead or contribute to acceptance. The tactical layer identifies tactics for influencing these psychological needs, while the gamification layer includes gamified solutions for engaging the user. These meta–models are instantiated with specific goals for a gamified design. For example, the Sufficient Acceptance (meta–)goal might be instantiated with a specific acceptance goal for a meeting scheduling system, such as "The system shall be used by $\geq$ 80% of its intended user community".

In summary, our aim is to provide the requirements analyst of social software systems with a framework that supports and guides her during acceptance requirements analysis, having as a final target the social software usage maximization. We propose that the analyst can use Agon either alone or in a participatory way by involving the stakeholders in the process. In fact, we think that Agon can be also useful for requirements elicitation. In our vision, The acceptance requirements analysis starts from the characterization of the user and uses AM for choosing psychological factors that are most suitable for the user indicated. Next, the TM is employed to refine these factors with tactics that can be operationalized through gamification and, finally, uses GM to determine a gamification solution.

III. The Models

Some statistics on our models: they consist of 270 goals, 376 relationships (refinements, operationalizations, positive/negative contributions). Full models are available online at [20]. Additionally, a glossary regarding elements of the Agon Framework is available online at [21]. In the following, we provide further details and examples regarding the elements of our framework.

Acceptance Model. This model is based on a number of acceptance models from the literature that have been proven useful in determining acceptance needs in different circumstances. Conducting a wide review, we selected the Unified Theory of Acceptance and Use of Technology (UTAUT) [9], the Technology Acceptance Model (TAM2) [10]), the Theory of Reasoned Action (TRA) [11], the Theory of Planned Behavior (TPB) [12], the Combined TAM and TPB (C-TAM-TPB) [13], etc. These models are part of an area of research that goes back more than 20 years and has had considerable impact on technology design practices. We integrated in AM their most valuable elements.

The full AM model [20] has `Sufficient Acceptance` as main need to be fulfilled. This goal receives contributions from less abstract needs such as `Create Facilitating Conditions` or `Improve Behavioral Intention` which in turn get contributions from `Reduce Effort Expectancy` and other lower level needs. Furthermore, the choice of which needs are selected for fulfilment is driven by the annotated CDRs, because for each kind of user there are correspondent techniques that enhance acceptance. For example, in Fig. 2, `Increase Social Influence` can be fulfilled if a user receives suggestions from people who

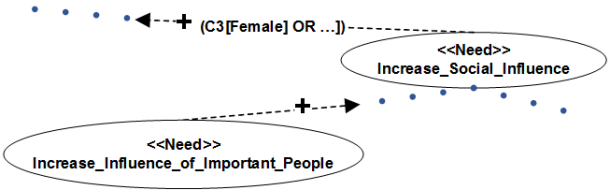

Fig. 2. Acceptance Need contributions annotated by CDRs.

are important (`Increase Influence of Important People`) for him/her (parents, friends, etc.), and has its maximum positive impact if the user has some characteristics, e.g., is female [9].

Tactical Model. TM captures alternative tactics for fulfilling acceptance requirements. For instance, the `Improve System Perception via IT` tactic [20] that refines `Improve Perceived Ease of Use`, could be operationalized, besides how it is in [20] by gamified training elements such as `Provide Tours` or `Provide Tutorials`, also by a serious game using a virtual simulated training. The latter, could be an alternative solution selected by TM and that comes from another possible model at the same level of GM, a serious game model. In this paper, we focus exclusively on gamification solutions, however, TM provides our framework with enough flexibility to add alternative solution models.

Gamification Model. We designed our GM [20] for operationalizating AM. We modelled GM on the baseline of theory and practice, gamification patterns, guidelines and real case studies in gamifying IT Systems [6], [7], [16], [22]. That allowed us to design a gamification concepts structure including positive interactions, synergies among concepts and avoiding in the model bad practices and incompatible combinations, offering best practices suggestions to the analyst and reducing the possibility of making analysis errors.

Moreover, we annotated the model by CDRs for selecting, in the reasoning phase, most suitable concepts on the basis of the context: the player characterization. In fact, also in the gamification case, for each kind of user there are correspondent techniques more suitable to maximize engagement. For instance, in Fig. 3, because a CDR claims that gamification social concepts have a positive impact in involving females [17], if we have female users and use the gamified market [6]

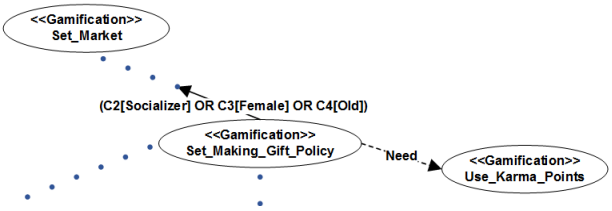

Fig. 3. Gamification Goal refinement annotated by a CDRs.

gamification goal (`Set Market`), it could be satisfied by a social gamification goal, `Set Making Gift Policy`, which means that a user can use accumulated karma points for making gifts to another person.

User Context Model and Context Dependant Rules. We designed our UCM [20] and annotated the models by CDRs, for providing Agon with a user characterization and rules for driving the reasoning, over the models, selecting the best design space elements for each abstraction layer in respect to the characterization indicated. We designed UCM by including strategic dimensions, proven in the literature [6], [9], [17], [18], that in the real life make difference in the way of people reacting to acceptance and gamification techniques. Dimensions we treat are: `Gender`, `Age`, `Player Type`, `Employment` and `Acceptance Subject`. For instance, one important aspect is Player Type, because different kinds of players are interested and stimulated by different types of gamification concepts. For managing it, we added in UCM the Bartle's player taxonomy [18] with 4 kinds of players (not mutually exclusive [20]): socializer, achiever, explorer and killer. Another important aspect is the Acceptance Subject (the system to be accepted) and its sub-dimensions: `Subject Awareness` is the user's familiarity regarding the acceptance subject and it is measured by `Expertise Level`; `Acceptance Voluntariness` regards the fact that the user is obliged or not to use the acceptance subject; `Precursor Subject` is related to the existence of a previous acceptance subject different to the proposed one.

IV. The Design Process, with an Example

Next, we illustrate our proposed framework with an extended version of the meeting scheduler exemplar that includes

some acceptance requirements. The full example can be found in [20]. Here, we sketch the steps of the design process and, in the last subsection, we briefly illustrate a summary of the full gamified solution [20] obtained by using our framework. Decisions taken by the analyst during the example, concerning choosing elements to keep/modify/discard, are made also on the basis of a glossary regarding elements of the Agon Framework [21].

Initial Requirements Model. We hypothesize that an analyst starts designing an initial goal model regarding a system: the Meeting Scheduler exemplar adapted to a Doodle like solution [20]. In a nutshell, main goals are: choosing potential participants, creating a Doodle defining possible dates, notifying participants asking they to compile the Doodle by filling preferred dates, etc. Thus, as in the meeting scheduler model [20], these goals are refined until the definitions of tasks, but in the case of `Convince Potential Participants to Compile Dates` we have a non–refined goal. This is a typical acceptance problem and in the following steps we show how Agon can support, guide and help the analyst in finding acceptance and gamification requirements maximizing the possibility to solve it.

Context Characterization. Firstly, the analyst gives in input to Agon a UCM instance representing the characterization of our participants group. It will guide the reasoning over abstraction layers. We specify `Group 1` [20] that is composed of old employed males that are achievers, have a low expertise level in using Doodle, are not obliged to compile the Doodle and have not used previous IT systems for scheduling meetings.

Context–Based Reasoning over Acceptance. Agon starts working on AM and, by using the UCM instance guiding the reasoning, produces an acceptance solution [20] having acceptance needs that best suit the context. For instance, for fulfilling `Improve Behavioral Intention`, we select `Reduce Effort Expectancy` and discard `Increase Social Influence`, because the latter has an annotation, as in Fig. 2, saying that it is better to adopt social influence techniques when users are females [9].

Furthermore, the analyst can select needs she prefers to maintain. For example, concerning `Increase Outcome Expectations`, the analyst may remove `Improve Skills` because it is an overkill to include training in the gamified solution for something as simple as meeting scheduling. Accordingly, the analyst keeps `Improve Perceived Ease of Use` to be achieved through a simple introduction to the tool. The chosen acceptance solution is added to the specification of the system-to-be.

Requirements Selection. On the basis of the acceptance solution, Agon shows tactics that can be used: `Support Achievement`, `Improve Perceived Status` and `Improve System Perception via IT`. The analyst may discard some of them reducing the solution space, however decides to keep all of them. Then, Agon finds strategies that can operationalize those tactics: `Design Gamification`, `Design Tangible Incentives` and `Design Serious Games`. The analyst keeps only gamification (in this work we focus exclusively on gamified solutions). Finally, Agon discovers that gamification can fulfil, besides tactics selected before, also other tactics that could be added in this phase to our specification. Moreover, additional tactics could contribute to engage users, but it is not guaranteed that they solve the acceptance problem. Additionally, the analyst, in all the phases, can decide if goals will be mandatory or preferences. In the example [20], the analyst adds `Fast Design`, `Low Cost Design`, `Increase User Surprise` and `Support User Penalization` to the specification.

Context–Based Reasoning over Gamification. In this phase, Agon works on the GM and uses the UCM to produce a gamified solution [20] having gamification goals that best suit context and other elements of the specification. For instance, Agon selects badges and for fulfilling `Set Kinds` [20], prefers `Set Publishable` because it is better to use publishable badges with elders. Furthermore, publishable badges are preferred because they operationalize `Improve Perceived Status`, one of ours mandatory requirements. Moreover, the analyst can select gamification goals she prefers to maintain or modify/substitute/add goals and tasks as we describe in the following.

Gamified Operationalization. The solution produced so far by Agon is a valuable specification including elements for maximizing acceptance, on the basis of user characterization indicated, through gamified strategies, but has still a little abstraction. In fact, this final step is required to create a final working solution having elements specific to the analyst's domain. The analyst uses the solution as a guidance including best gamification practices and she has just to complete it with elements specific of her domain. Thus, the analyst has to select gamification goals she prefers to maintain, optionally modify/substitute they and, above all, operationalize they by adding goals and tasks (hexagons in [20]). For example, as it is shown in the final solution [20], the analyst, concerning leader–boards and in particular `Set Leader-boards Calculation Strategy`, removes `By Point Systems` and adds another strategy: `By Compiling End Time`. This is because Agon proposes to calculate the ranking by a generic best practice (by points) and the analyst chooses a specific strategy that better fits the meeting scheduler domain and could not be inferred by Agon: compute the leader–board on the basis of who fills before the Doodle.

The Gamified Meeting Scheduler in a Nutshell. This is the summary of the full gamified solution [20] obtained by using Agon for solving the non–refined `Convince Potential Participants to Compile Dates` goal of the meeting scheduler.

In the game designed, potential participants can have a tour of the system features and compile the Doodle (Gamifiable Actions). The tour is proposed before compiling (Propose Tour Before Compiling) and optionally can be skipped (Set Skip The Tour feature). If the tour is completed, the user acquires expertise concerning the system and win a badge representing it (Win Doodle Meeting Scheduler Expert Badge). Actually, this expertise is more a perceived knowledge, because she just sees a demo describing main features of the system, without having, for instance, a complex training (Define Training Paths) that could lead to have a deep knowledge of a system. In this case it is enough, because the analyst desires a solution with soft training. In fact, in the different phases, she has confirmed the employment of the Improve Perceived Ease of Use need that is refined by the Improve System Perception via IT tactic which in turn is operationalized by soft training such as the Provide Tours gamification goal. Instead, if it was preferred an hard training, it would be selected the Support Skill Improvement tactic that is operationalized by Define Training Paths, which needs Provide Tutorials, namely active tasks with well-defined learning paths.

Continuing the game description, each person that compiles the Doodle wins 10 Redeemable Points (Set 10 RP) and a badge (Set Potential Participant Badge): the first one wins the badge Set First Compiling Badge, the second one Set Second Compiling Badge and the third one Set Third Compiling Badge. Contextually, these 3 winners are the same of the podium (Set Traditional Podium) of the leader–board defined (Set First Doodle Compilers LB) and they achieve respectively: Win 10000 RP Points, Win 1000 RP Points and Win 100 RP Points. There is also a market where tangible rewards (Add Tangible Rewards) can be redeemed, by cumulated redeemable points, thanks to the redeeming policy (RP Define Exchange Points Rewards) defined for the market policies (Set Market Policies). Moreover, all the badges are publishable, as expressed through Set Publishable, and this operationalizes one of our specification tactic: Improve Perceived Status. Finally, badges can be published in a community (Set Community) having different social activities.

V. Discussion

Novelty of the Framework. Acceptance requirements for software systems are nearly as well understood as other well–studied requirements, such as performance, security and usability. Our review of the literature suggests that so far, acceptance requirements have been treated either in a more general context as technology acceptance, with many applications reported in the literature [1]–[4], or for very specific cases concerning software systems, as in [23]. The novelty of our proposed framework rests on its focus to social software where more general technology acceptance techniques may or may not apply. In fact, our models are generic reference models related to different layers of abstraction. Specifically, the AM concerns psychological factors influencing acceptance, while GM captures gamified best practices for different classes of users. However, we do not exclude that our framework could be valuable also in more general contexts (e.g., technology acceptance).

What Comes Next. To complete this ongoing research, we propose to extend and improve our models, develop a prototype Agon tool, and conduct a thorough evaluation. Specifically, we plan to review other acceptance models and integrate them in our AM and GM. At the same time, we will continue to improve the UCM by adding other dimensions relevant to acceptance requirements and gamification. Secondly, we are developing an environment to support the exploration of acceptance and gamification alternatives during design. That prototype will support the Agon process, paying particular attention to usability, given that Agon models are large and will get even larger. Finally, we plan to evaluate Agon with more complex real case studies in heterogeneous fields in order to confirm its generality, versatility and utility. We plan to evaluate also the balance between the cost of applying the approach and the acceptance ratio gained by using the resulting gamification.

VI. Conclusion

Social software is successful only when it is used. Accordingly, the requirements for any such software system must include acceptance requirements. We are developing Agon, a framework for systematically dealing with acceptance requirements by using gamification techniques. The framework includes a meta–model of acceptance requirements, alternative tactics for fulfilling them, and gamification solutions to these tactics. The meta–model is based on a comprehensive review and evaluation of the literature. In addition, the framework includes a design process for incrementally generating a gamified solution for a specific system by instantiating the meta–model. We illustrate the framework with the Meeting Scheduler exemplar adapted to a Doodle-like solution.

Gamification constitutes one approach to operationalizing acceptance requirements. One direction for future research is to extend the framework by considering other approaches, such as serious games, game metaphors and tangible incentives. The effectiveness of a gamified solution to an acceptance requirement may vary over time. For instance, users may get bored with a game played over meeting scheduling. Accordingly, we also envision to make our gamified solutions adaptive, so that they monitor their performance relative to acceptance requirements and reconfigure their games accordingly if user acceptance drops. Finally, our context model could be improved by exploring additional dimensions that further contextualize gamified solutions to acceptance requirements.

References

[1] J. Beer and L. Takayama, "Mobile Remote Presence Systems for Older Adults: Acceptance, Benefits, and Concerns," in *Proc. of the 6th international conference on Human-robot interaction*. ACM, 2011.

[2] K. Arning, B. Trevisan, M. Ziefle, and E. Jakobs, "Eliciting User Requirements and Acceptance for Customizing Mobile Device System Architecture," in *Design, User Experience, and Usability. Design Philosophy, Methods, and Tools*. Springer, 2013, pp. 439–448.

[3] F. Davis and V. Venkatesh, "Toward Preprototype User Acceptance Testing of New Information Systems: Implications for Software Project Management," *IEEE Transactions on Engineering Management*, 2004.

[4] R. Poston and A. Calvert, "Vision 2020: The Future of Software Quality Management and Impacts on Global User Acceptance," in *HCI in Business*. Springer, 2015, pp. 748–760.

[5] M. Kaptein, P. Markopoulos, B. De Ruyter, and E. Aarts, "Personalizing Persuasive Technologies: Explicit and Implicit Personalization Using Persuasion Profiles," *Intern. Journal of Human-Computer Studies*, 2015.

[6] G. Zichermann and C. Cunningham, *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. "O'Reilly", 2011.

[7] J. Schell, *The Art of Game Design: A book of lenses*. CRC Press, 2014.

[8] L. Piras, G. Valetto, A. Marconi, and M. Pistore, "Virtual Coaches for Mission-Based Gamified Smart Communities," in *CHI-PLAY 2015 Workshop, Personalization in Serious and Persuasive Games and Gamified Interactions*. ACM, 2015.

[9] V. Venkatesh, M. Morris, G. Davis, and F. Davis, "User Acceptance of Information Technology: Toward a Unified View," *MIS quarterly*, 2003.

[10] V. Venkatesh and F. Davis, "A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies," *Management science*, vol. 46, no. 2, pp. 186–204, 2000.

[11] B. Sheppard, J. Hartwick, and P. Warshaw, "The Theory of Reasoned Action: A Meta-Analysis of Past Research with Recommendations for Modifications and Future Research," *Journal of Cons. Research*, 1988.

[12] I. Ajzen, "The Theory of Planned Behavior," *Organizational Behavior and Human Decision Processes*, vol. 50, no. 2, pp. 179–211, 1991.

[13] S. Taylor and P. Todd, "Assessing IT Usage: The Role of Prior Experience," *MIS quarterly*, pp. 561–570, 1995.

[14] J. Mylopoulos, L. Chung, and B. Nixon, "Representing and Using Nonfunctional Requirements: A Process-Oriented Approach," *IEEE Transactions on Software Engineering*, vol. 18, no. 6, pp. 483–497, 1992.

[15] L. Chung, B. Nixon, E. Yu, and J. Mylopoulos, *Non-Functional Requirements in Software Engineering*. Springer, 2012, vol. 5.

[16] J. Hamari, "Do Badges Increase User Activity? A Field Experiment on the Effects of Gamification," *Computers in Human Behavior*, 2015.

[17] J. Koivisto and J. Hamari, "Demographic Differences in Perceived Benefits from Gamification," *Computers in Human Behavior*, 2014.

[18] R. Bartle, "Hearts, Clubs, Diamonds, Spades: Players Who Suit MUDs," *Journal of MUD Research*, 1996.

[19] G. Orsi and L. Tanca, "Context Modelling and Context-Aware Querying," in *Datalog Reloaded*. Springer, 2011, pp. 225–244.

[20] L. Piras, P. Giorgini, and J. Mylopoulos. Full Models of this work. [Online]. Available: https://pirasluca.wordpress.com/home/acceptance/

[21] ——. Glossary regarding elements of the Agon Framework Models. [Online]. Available: https://pirasluca.wordpress.com/home/acceptance/glossary/

[22] R. Kazhamiakin, A. Marconi, M. Perillo, M. Pistore, G. Valetto, L. Piras, F. Avesani, and N. Perri, "Using Gamification to Incentivize Sustainable Urban Mobility," in *1st Intern. Smart Cities Conf. (ISC2)*. IEEE, 2015.

[23] A. Sutcliffe, P. Rayson, C. Bull, and P. Sawyer, "Discovering Affect-Laden Requirements to Achieve System Acceptance," in *22nd International Requirements Engineering Conference (RE)*. IEEE, 2014.