Universidade Federal da Bahia

Universidade Estadual de Feira de Santana

DISSERTAÇÃO DE MESTRADO

Gamifying user interaction to increase collaboration:

The G.A.M.E. conceptual framework

Jailson Aldérico de Brito Júnior

Mestrado Multiinstitucional em Ciência da Computação

MMCC

Salvador – BA

2015

JAILSON ALDÉRICO DE BRITO JÚNIOR

GAMIFYING USER INTERACTION TO INCREASE COLLABORATION: THE G.A.M.E. CONCEPTUAL FRAMEWORK

Dissertação apresentada ao Mestrado em Ciência da Computação da Universidade Federal da Bahia e Universidade Estadual de Feira de Santana, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Vaninha Vieira dos Santos
Co-orientador: Adolfo Almeida Duran

Salvador
23 de Julho de 2015

Ficha catalográfica.

DE BRITO JUNIOR, Jailson Aldérico

Gamifying user interaction to increase collaboration: The G.A.M.E. conceptual framework/ Jailson Aldérico de Brito Júnior– Salvador, 23 de Julho de 2015.

126p.: il.

Orientador: Vaninha Vieira dos Santos.
Co-orientador: Adolfo Almeida Duran.
Dissertação (Mestrado)– Universidade Federal da Bahia, Instituto de Matemática, 23 de Julho de 2015.

1. Sistemas colaborativos. 2. Crowdsourcing. 3. Gamificação. 4. Interação de Usuário. .
I. DOS SANTOS, Vaninha Vieira. II. DURAN, Adolfo Almeida.
III. Universidade Federal da Bahia. Instituto de Matemática. IV. Título.

TERMO DE APROVAÇÃO

JAILSON ALDÉRICO DE BRITO JÚNIOR

GAMIFYING USER INTERACTION TO INCREASE COLLABORATION: THE G.A.M.E. CONCEPTUAL FRAMEWORK

Esta dissertação foi julgada adequada à obtenção do título de Mestre em Ciência da Computação e aprovada em sua forma final pelo Mestrado em Ciência da Computação da UFBA-UEFS.

Salvador, 23 de Julho de 2015

Profa. Dr. Adolfo Almeida Duran (orientador)
Universidade Federal da Bahia

Prof. Dr. Frederico Araújo Durão
Universidade Federal da Bahia

Profa. Dra. Patricia Cabral de Azevedo Restelli Tedesco
Universidade Federal de Pernambuco

To my wife Rosara

ACKNOWLEDGEMENTS

I would never have been able to finish my dissertation without the guidance of my advisors, the help of my friends, and the support of my family and wife.

I wish to express my sincere thanks to my advisors Vaninha and Adolfo. I am extremely thankful and indebted to them for sharing their expertise and valuable guidance and encouragement extended to me. I would also like to thank the evaluation of the professors Patricia Tedesco and Fred Durão, who enriched the final version of this work.

I am also grateful to many friends and colleagues who helped in someway during this journey. Special thanks go to Amancio Santos and Luciana Lins with their enlightenment on qualitative evaluations.

I take this opportunity to express my gratitude to all the participants of the evaluations performed in this work, which I will not name here due to privacy.

This work was partially supported by CNPq (process 560135/2010-6), RNP-CTIC (SIMTUR project) and the National Institute of Science and Technology for Software Engineering (INES[1]), funded by CNPq and FACEPE, grants 573964/2008-4 and APQ-1037-1.03/08.

Finally, I would like to thank my wife, Rosara. She was always there cheering me on and giving me power-ups to complete this adventure.

[1] http://www.ines.org.br

Brave, survive a crazy world
struggling not to fall,
trying to be the kind of mind
that wants to fly high.

—DANNY NASCIMENTO

RESUMO

Sistemas colaborativos são sistemas baseados em computador que apoiam grupos de pessoas em atividades que demandam colaboração. Um meio de colaboração é o *crowdsourcing* - o ato de tomar tarefas tradicionalmente feitas por um individuo e terceiriza-las para um grande grupo de pessoas. Comunidades online utilizam *crowdsourcing* para diversos fins, através de uma multidão de usuários que colaboram para resolver problemas. Massa crítica é obrigatória para o sucesso do *crowdsourcing*. Além de massa crítica, *crowdsourcing* depende da motivação e do comportamento dos participantes.

É necessário entender a motivação do usuário em participar, e os desenvolvedores precisam incorporar elementos de motivação e projetar mecanismos de incentivo. Um dos desafios do *crowdsourcing* reside no projeto de interação de usuário que atraiam e mantenham um grande número de pessoas participando.

Gamificação, uso de elementos de jogos em contextos que não são jogos, tem sido aplicada para apoiar o engajamento de pessoas em comunidades online que demandam colaboração dos participantes. Gamificação aproveita-se de como os jogos eletrônicos engajam jogadores a resolverem problemas fictícios para engajar os mesmos jogadores a resolver problemas do mundo real. Apesar dos estudos sobre gamificação em sistemas colaborativos estarem crescendo, a maioria das abordagens experimentais de gamificação são *ad-hoc*. Não existe uma abordagem sistemática para implementar gamificação em sistemas colaborativos.

Esse trabalho tem o objetivo de investigar como a gamificação pode ser utilizada no projeto de interação de usuário para aumentar a colaboração em sistemas *crowdsourcing*. Nessa dissertação, nós apresentamos as nossas descobertas e discutimos as decisões tomadas na gamificação em nosso estudo de caso. Propomos uma abordagem sistemática (denominada G.A.M.E.) para apoiar o projeto da interação de usuário para aumentar a colaboração em sistemas *crowdsourcing*.

Além disso, nós apresentamos um estudo qualitativo sobre o uso do G.A.M.E. por diferentes desenvolvedores. O estudo qualitativo nos permitiu coletar impressões do ponto de vista de profissionais de diferentes áreas. Dos resultados alcançados com essa avaliação nós obtemos evidencias da utilidade da abordagem G.A.M.E., e os aspectos críticos do projeto de gamificação para sistemas colaborativos.

Palavras-chave: Sistemas colaborativos, Crowdsourcing, Gamificação, Interação de Usuário.

ABSTRACT

Collaborative software, also known as groupware, supports groups in collaborative activities. Groupware are *computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment.* A means for collaboration is crowdsourcing - the act of taking tasks traditionally done by an individual and outsourcing it to a large group of people. Online communities use crowdsourcing for several purposes. Crowdsourcing systems call a crowd of users to collaborate and solve problems. Critical mass is mandatory to the success of crowdsourcing. In addition to having a mass of users, crowdsourcing also depends on people's motivation and behavior in participating.

It is necessary to understand the user's motivation to participate, and designers should incorporate motivation elements to provide incentive mechanisms. One main crowdsourcing challenge is how to design user interfaces that can attract and sustain many people to joining in crowdsourcing.

Gamification, the use of game elements in non-game contexts, has been used to engage people in online communities that require collaboration from the participants. It takes advantage of how games engage players to solve fictional problems in fictional worlds to engage the same players to solve real world problems. Although gamification in collaborative software studies have been growing, there is no systematic approach to implement gamification in collaborative software.

This work aims to investigate how gamification can be used to design user interaction aiming to increase collaboration in crowdsourcing systems. In this dissertation, we present our findings and discuss our decision making for gamifying our case study. We propose a systematic approach (named G.A.M.E.) to support designing user interaction of collaboration in crowdsourcing systems.

Moreover, we present a qualitative study on the usage of G.A.M.E. by different developers. The qualitative study allowed us to gather impressions from point-of-view of professionals from different areas. From the results achieved with this evaluation we obtained evidence of the usefulness of the G.A.M.E. approach, and critical aspects of gamification design for collaborative software.

Keywords: Collaborative Software, Crowdsourcing, Gamification, User Interaction.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Collaboration involves communication, coordination and cooperation (Pimentel et al., 2006): communication is done by messages exchanges; coordination is achieved through management of people, tasks and resources; and cooperation is achieved through transactions in a shared space for execution of tasks. A means for collaboration is crowdsourcing.

Crowdsourcing has become very popular since the term was firstly coined by Howe (2006) to describe the phenomena of companies using a workforce around the world to execute tasks as long as they are connected to the Internet. Crowdsourcing can also be defined as a distributed problem-solving approach using knowledge, energy and creativity from an online community (Bell, 2013; Parvanta; Roth; Keller, 2013). Communities are groups of people who share common interests, beliefs and values and who interact in some mediated way (Cullen; Morse, 2011). Online communities use crowdsourcing for several purposes (Doan; Ramakrishnan; Halevy, 2011). Crowdsourcing systems call a crowd of users to collaborate and solve problems explicitly or implicitly (Doan; Ramakrishnan; Halevy, 2011). When the collaboration is explicit (people perform actions on the system), crowdsourcing systems depend on user participation to accomplish the tasks. Critical mass is mandatory for the success of crowdsourcing (Russ, 2007). In addition to having a critical mass of users, crowdsourcing also depends on people's motivation and participation (Zhao; Zhu, 2012).

In online communities, most users do not contribute (Hill et al., 1992). Nielsen (2006) states that most online communities work following the trend 90-9-1, where: 90% of users just lurk and never contribute; 9% contribute casually or hastily; and just 1% of users are responsible for most of the action within the community. This phenomenon is known as "Participation Inequality" and it is impossible to overcome it (Nielsen, 2006).

Studies about participation encouragement in online communities are related with user experience enhancement, for example, easing contributions or turning participation a side effect (Nielsen, 2006). Other approaches try to enhance the user intrinsic interest in tasks by providing extrinsic rewards, or by increasing the expected benefits that will come through individual contributions to the group outcomes (Kraut; Resnick, 2011; Knautz

et al., 2012; Liu et al., 2011). As stated by Lu (2011), improvements in usability and social development result in usefulness, enjoyment and a sense of belonging of members, which promotes continuous participation .

Investigating the behavior of crowdsourcing participants is an important research topic (Zhao; Zhu, 2012). Even though mass collaboration seems to be self-organized, it is necessary to introduce mechanisms that will ensure efficiency and promote a proper outcome (Potter; Mcclure; Sellers, 2010). It is also important to explore incentive mechanisms which can help the online community aggregate the mass of users and gain valuable solutions from them (Zhao; Zhu, 2012). Individual motivation is needed for effective mass collaboration, but the most important thing is the individual's interest in the products of the collaboration (Potter; Mcclure; Sellers, 2010). It is necessary to understand the user motivation to participate, and designers should incorporate motivation elements in the design of incentive mechanisms (Zhao; Zhu, 2012). Pan and Blevis (2011) point out that one of the main crowdsourcing challenges is how to design user interfaces that can attract and sustain numerous people in crowdsourcing.

Another strategy that has been used to promote user participation is the use of game design techniques. Games have successful strategies to grant enjoyable user experience (Deterding, 2012). Digital marketing and social media professionals have been using this approach under the name of gamification (Liu et al., 2011). Gamification has been defined as the thinking process based on games and its mechanisms to get users to solve problems (Zichermann; Cunningham, 2011). Gamification is also defined as the use of game design elements in non-game contexts (Deterding et al., 2011). The game design involves all the steps from the early conception to the refining of the game requirements and specifications and how the player is supposed to interact with it. So the game designers conceive how game design elements will combine to drive the player to the game experience. Game design elements include scoring systems, rewards, levels and rankings (Zichermann; Cunningham, 2011).

1.1 MOTIVATION

The interest in gamification has been increasing for many reasons (Deterding, 2012), however, it is shrouded in controversies. For example, SessionM[1], a digital media company, after some A/B testings, claimed that their gamification tools helped online applications to increase 35% of user retention and increased user engagement by 250% (Sessionm, 2012). On the other hand, Gartner Inc.[2], IT research and advisory company, is pessimistic about gamification. According to their analysis in 2012, by 2014 80% of current gamified applications would fail to meet business objectives primarily due poor gamification design (Gartner, 2012). This scenario suggests that it is not clear how gamification works and how to implement successful gamification in online applications.

The game design of most gamified applications is limited to obvious game mechanics (such as points, badges and leaderboards) instead of designing player-centric applications with important game design elements such as meaningful game economy or balancing

[1]http://www.sessionm.com - A digital media company

[2]http://www.gartner.com - Gartner Inc. - an IT research and advisory company

of competition and collaboration (Deterding, 2012). Most gamification design guidelines (Dubois, 2012; Liu et al., 2011; Zichermann; Cunningham, 2011; Duggan; Shoup, 2013; Nicholson, 2012) concerns only on introducing game elements to the application design. These studies present gamification as an environment surrounded by point system, challenges, victory conditions, rewards, rankings, medals, status and social networks. While marketers defend gamification as an engagement driver, game designers argue that the current implementations of gamification based on points, badges and leaderboards are taking the elements that are least essential to games and representing it as the core of the experience (Deterding, 2012).

Despite some famous crowdsourcing systems using gamification such as StackOverFlow[3] and TripAdvisor[4], there are uncertainties about how to implement gamification in collaborative software (Bastos et al., 2013). Although the number of studies into gamification in collaborative software has been increasing every year, the majority of gamification experimental approaches are ad-hoc and there is no consolidated method, model, process or conceptual framework to implement gamification in collaborative software (Bastos et al., 2013).

1.2 OBJECTIVES AND RESEARCH QUESTIONS

This research aims to investigate how the systematic use of game design elements could improve user interaction in crowdosourcing systems from the early design stages to the final evaluation. The main goal of this project is:

- MQ: to develop a systematic approach that can be used by developers to design gamification for collaborative systems with the use of game design elements.

Our approach is innovative by identifying significant relationships between game design techniques and software engineering practice. To achieve the main goal of this research, we expect to answer the following Research Questions (RQ):

1. RQ1: What are the main aspects to consider in the design of gamification for user interaction in crowdsourcing systems?

2. RQ2: How can these aspects be shaped into a systematic approach for software design?

3. RQ3: Would such a systematic approach be useful for the design of gamification for crowdsourcing systems?

1.3 METHODOLOGY

In order to answer RQ1 it was necessary to understand the state-of-the-art of collaborative software, user interaction, gamification and how each concept relates to others. We answer RQ2 by proposing a systematic approach and designing a crowdsourcing system as a case

[3]http://www.stackoverflow - Q&A site for programmers
[4]http://www.tripadvisor.com - Consumers reviews about flights, hosting, trips and vacations.

study. Through qualitative evidence gathered from professionals using the proposed approach it was possible to answer RQ3.

To understand how to use gamification in crowdsourcing systems, we adopted a bottom-up and top-down approach: from specific applications to derive an abstraction or method. In this sense, we chose Wikibus, a collaborative public transportation system information repository (Moitinho, 2013), which use social networks and crowdsourcing techniques for: (i) collaborative editing of bus routes; (ii) occurrence registration using mobile devices; and (iii) incident reporting and viewing on the web. To implement gamification in Wikibus we consider our application as a game instead of an ordinary application. If we consider a collaborative software as a game we can use the available and consolidated methods and processes to design good, amusing and successful games.

G.A.M.E. (an acronym for its 4 phases Gathering, Analysis, Modeling and Execution) provides a process that combines knowledge from software engineering, collaborative software, game design and interaction design. The purpose of G.A.M.E. is to support turning an ordinary user experience into a gameful experience - with qualities of gaming. G.A.M.E. was built to be used in the interaction design of collaborative tasks for crowdsourcing systems.

To evaluate the G.A.M.E. approach we performed a qualitative study with other developers which allowed us to gather data from the point-of-view of software development. From the results of this evaluation we obtained evidence of the usefulness of our approach and its most critical aspects.

1.4 DISSERTATION STRUCTURE

The organization of the remaining of this dissertation is illustrated in Figure 1.1. The contents of the chapters are detailed in the following.

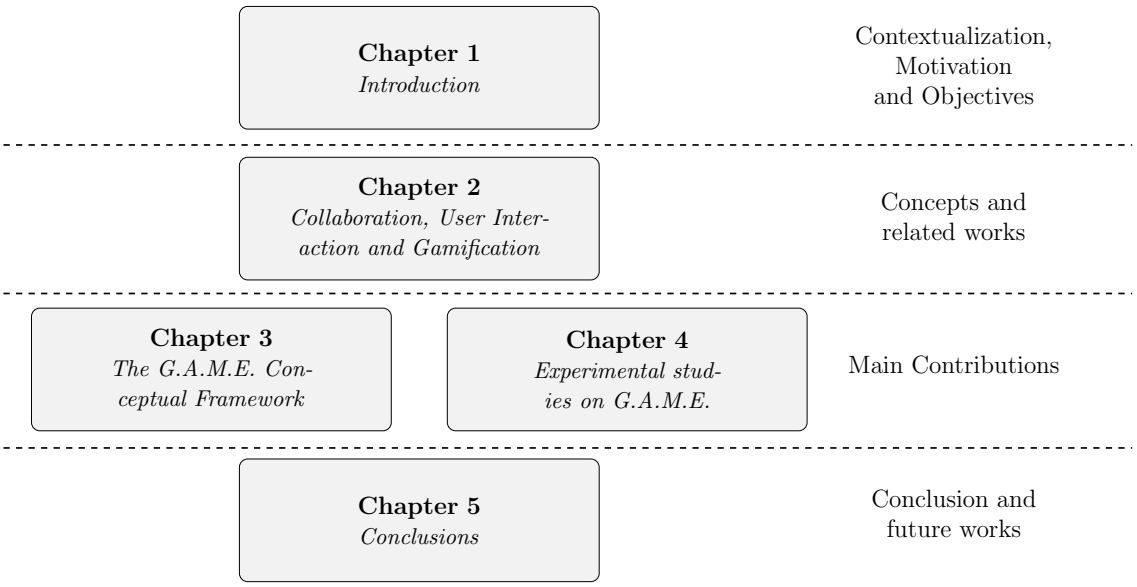


Figure 1.1: Dissertation structure.

Chapter 2 presents an overview of the basic concepts used in this work: Collaboration and Crowdsourcing as well as Gamification, Interaction Design and Game Design. It also explores the state of the art of Gamification in Collaboration Software, discussing their characteristics, strengths and weaknesses and identifying possible improvements.

Chapter 3 presents G.A.M.E., a conceptual framework to guide developers in designing gamification into crowdsourcing systems.

Chapter 4 presents the experimental studies performed to assess the usefulness and potential of the G.A.M.E. framework. We discuss how G.A.M.E. was used to design gamification in a case study: the Wikibus. Then, we present how we conducted an experiment with software development professionals to investigate how G.A.M.E. was used and useful in designing gamification for crowdsourcing systems using a systematic approach.

Finally, Chapter 5 summarizes the proposed work and discusses the contributions achieved, indicating some directions in which the presented research could be extended.

Chapter 2

COLLABORATIVE SOFTWARE, INTERACTION DESIGN AND GAMIFICATION

This chapter presents the key concepts for this work. Collaborative software provides means for people to work together towards a common goal. A way to allow collaboration is through crowdsourcing. There are several challenges when designing crowdsourcing applications. An important issue is how to recruit and retain users (Doan; Ramakrishnan; Halevy, 2011). The user interfaces must be designed in such a way as to attract and sustain a mass of users, with different profiles and background, to work together (Pan; Blevis, 2011). One strategy to increase user participation is the use of game design techniques, as games have successful strategies to grant an enjoyable experience (Deterding, 2012). Digital marketing and social media professionals have been using this approach under the name of gamification (Liu et al., 2011). Section 2.1 overviews collaborative software and crowdsourcing. Section 2.2 discusses user interaction and related concepts. Section 2.3 presents a big-picture of gamification.

2.1 COLLABORATIVE SOFTWARE

Collaborative software, also called groupware, is defined as *"computer-based system that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment"* (Ellis, 1991). From this definition, two concepts are particularly important to distinguish collaborative software from other kinds of software: common task and shared environment. Another difference is that ordinary systems support a user to interact with a system and groupware support user-to-user interaction through three key areas: communication, cooperation, and coordination (Ellis, 1991; Fuks; Raposo; Gerosa, 2003).

2.1.1 3C Collaboration Model

Communication is necessary when people need to work together. In groupware, participants negotiate and settle commitments that change the shared environment states and

affects future actions (Fuks; Raposo; Gerosa, 2003). Groupware must support communication by managing states transitions, events and the commitments of each participant (Fuks; Raposo; Gerosa, 2003) and the challenge is to use the proper tools to promote interactions that will benefit the most of the media (Ellis, 1991).

Groupware communication generates commitments. To assure the fulfillment of these commitments and the accomplishment of tasks, some coordination is necessary (Fuks; Raposo; Gerosa, 2003). The coordination problem is how to harmoniously integrate the individual efforts toward the accomplishment of a common goal (Ellis; Gibbs; Rein, 1991). In some collaborative software, coordination is the responsibility of the social protocol, where there is no explicit mechanism of coordination. However, some tasks require explicit coordination mechanisms to assure the collaboration success (Fuks; Raposo; Gerosa, 2003). We can divide coordination into three phases. The first phase relates to actions required to prepare the collaboration, normally concluded before the cooperation work starts, e.g., identification of goals, mapping of goals in tasks, participant selection and distribution of tasks among participants (Fuks; Raposo; Gerosa, 2003). The second phase is the management while people are performing the tasks. The last phase happens when the tasks are finished and involves the evaluation of performed tasks and sometimes it is also necessary to document the collaboration process (*post-mortem*) (Fuks; Raposo; Gerosa, 2003).

Cooperation is about people working together or individually towards the common goal of a group (Fuks; Raposo; Gerosa, 2003). Cooperation is the operation of group members in a shared environment to perform tasks managed by the coordination (Fuks; Raposo; Gerosa, 2003). Each participant acts individually and the sum of all individual efforts results in the cooperation outcome of the groupware.

Awareness of collaboration pervades communication, coordination and cooperation. All the kinds of interaction that occur in the group generate awareness, mediating all collaboration steps. Through awareness, individuals have knowledge about the useful information, for example: the common goal, the work progress, what to do, how to proceed, what is the impact of their actions, boundaries, who is closer (Fuks; Raposo; Gerosa, 2003).

The 3C Collaboration model (Fuks et al., 2005) summarizes how collaborative work occurs. Figure 2.1 shows the 3C Model that organizes the four discussed concepts: Cooperation; Coordination; Communication; and Awareness. Collaboration depends on communication between individuals with some Coordination to perform Cooperation; The 3Cs foster awareness that users have of Collaboration; And awareness also mediates each 'C'.

Another collaboration model is the 4C Collaboration Model (Costa; Loureiro; Reis, 2014) which is based on almost all the same principles as the 3C Collaboration Model. The main contribution of the 4C Collaboration Model is that it covers cooperation as two dimensions: collaboration and cooperation. The collaboration happens when the participants work together with other participants. The cooperation is the collaborative work that is done individually by each participant.

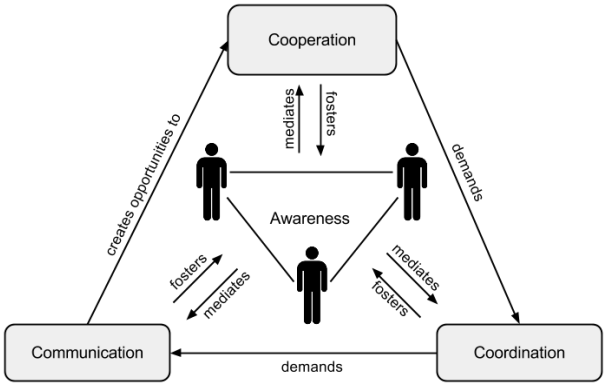


Figure 2.1: 3C Collaboration Model (Fuks et al., 2005).

2.1.2 Crowdsourcing

A way to make a large number of people work together collaboratively is through crowdsourcing. Originally defined in 2006 by Howe (2006), crowdsourcing is the act of taking tasks traditionally done by an employee and outsourcing them to a large group of people or a community through an open call. Crowdsourcing can also be used to engage people with different talents, knowledge and perspectives to create and disseminate products and services (Parvanta; Roth; Keller, 2013).

Crowdsourcing-based systems promote what has been called "citizen power", stimulating people to help solve public and governmental problems (Vieira; Caldas; Salgado, 2011). Several kinds of applications can take advantage of crowdsourcing, including image and video tagging (Freiburg; Kamps; Snoek, 2011; Fuglestad et al., 2012), reporting crimes (Furtado et al., 2010) and urban mobility (Vieira; Caldas; Salgado, 2011). Paid crowdsourcing markets such as Amazon Mechanical Turk uses the workforce of a crowd for tasks such as content creation, transcription or web research (Kaufmann; Veit, 2011). When a task is easier to solve by humans than by computers or if it would take too long or cost too much for a single individual to solve it, it may be worth crowdsourcing it. Bell (2013) points out some benefits of crowdsourcing for organizations:

- Low cost and high speed to solve problems;
- Payments are upon results;
- Variety of talents involved;
- Users may experience a close relationship with organization which results in a sense of belonging and contribution through collaboration.

An important issue related to crowdsourcing is how to recruit and retain users (Doan; Ramakrishnan; Halevy, 2011). Russ (2007) addresses this issue through a process of three stages: initiation, propagation, and amplification. Figure 2.2 shows a simplified phase model of online crowds. In the initiation step it is important to have the best content to attract users. When you achieve a good collection of content, then you should allocate

efforts to retain and recruit the best members to achieve critical mass. Then, you should work in facilitating socialization through the community causing a chain reaction that can maintain high participation over the time.

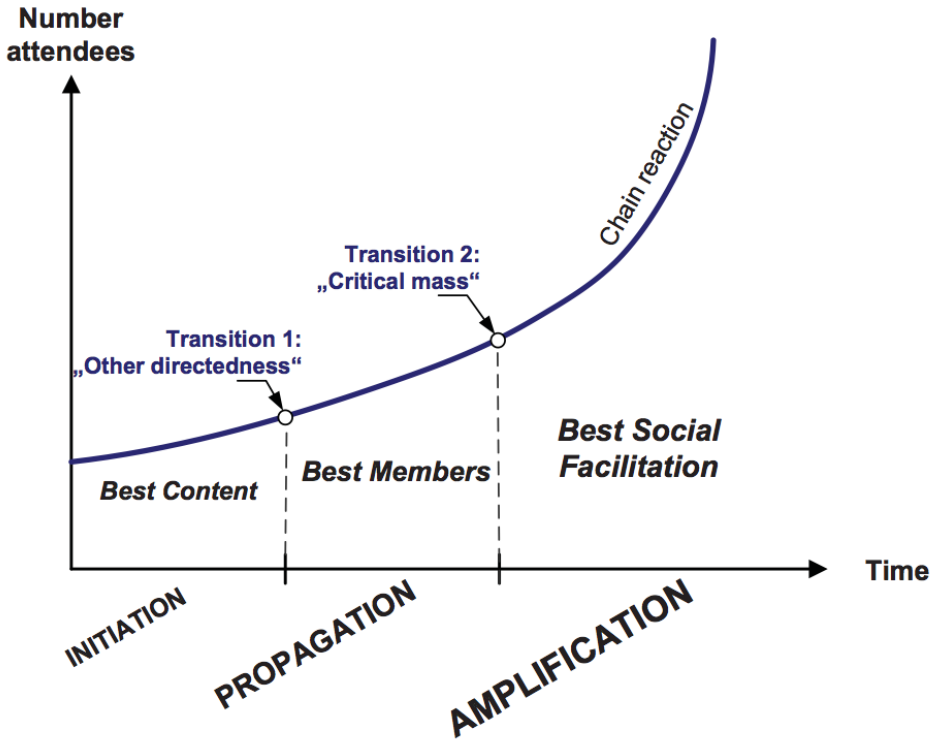


Figure 2.2: Simplified phase model of Online Crowds (Russ, 2007).

In Section 2.1, we discussed the main elements of collaborative work: communication, coordination, cooperation and awareness. In crowdsourcing systems these elements are also present. Crowdsourcing systems provide a means to exchange messages and information between participants (Peng; Babar; Ebert, 2014). Depending on the crowdsourcing task, there are people acting as providers and others as requesters. Providers negotiate with requesters about requirements and terms through the exchange of information and opinions, and requesters need to be aware of provider capability, experience, and reputation (Peng; Babar; Ebert, 2014). Another point concerning crowdsourcing that affects communication is its distributed nature. Crowd participants are usually geographically distributed, sometimes worldwide, and need to communicate to solve issues, cooperate and coordinate tasks (Peng; Babar; Ebert, 2014)

Getting a crowd of people to work towards a common goal requires a coordination. Crowdsourcing systems must enable the management of people and processes to facilitate creating, assigning, executing, evaluating, and rewarding crowdsourced tasks while supervising the commitments of both requesters and providers (Peng; Babar; Ebert, 2014). For example, crowdsourcing must provide a means to solve conflicts between providers and requesters.

The cooperation of a crowdsourcing system deals with the challenge of numerous

collaborations from different people in diverse places. A crowdsourcing system should support various cooperation tasks by providing the facilities for a shared environment and encouraging user interactions with artifacts synchronously or asynchronously (Peng; Babar; Ebert, 2014).

Group awareness lets group participants obtain knowledge of the working context by understanding the processes, tasks and project status (Peng; Babar; Ebert, 2014)

2.1.3 Evaluating Collaboration

The Systemic Evaluation for Stakeholder Learning (SESL) method (Ramage, 1999) can be used to evaluate collaborative software. A way to instantiate SESL is through the Five Step Method (Ramage, 1999) as illustrated in Figure 2.3. The first step consists of determining the type and purpose of the evaluation. Then, execute the study & analyze cycle through key issues. Finally, this encourage the learning for stakeholders by reporting the results.

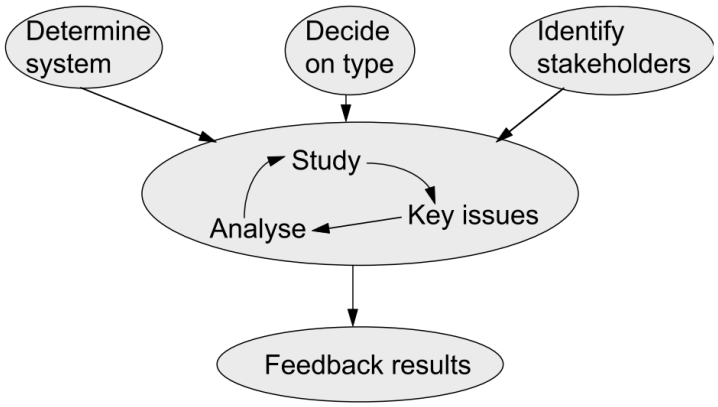


Figure 2.3: SESL - Five Step Method (Ramage, 1999).

There are eight layers of evaluation from SESL as represented in Figure 2.4. We should perform the evaluation from the inner layers to the outer layers. The most inner level is functionality, it evaluates if the application does what it was supposed to do. Efficacy goes beyond by assessing if the application performs well. The usability layer examines how easy, intuitive and useful the approach is. Standards checks if what was done complies with known standards. The last four layers are about how the collaborative software impacts the individual, the group, the organization and the society, respectively.

Each layer of evaluation consists of a key question, to evaluate all possible aspects of a collaborative software. However, according to (Ramage, 1999) most evaluation practices do not reach beyond the fourth layer. Ramage (1999) gives examples of how we should formulate the Eight Key Questions for evaluation:

1. Does it work? *(functionality)*

2. Does it work well enough? *(efficacy)*

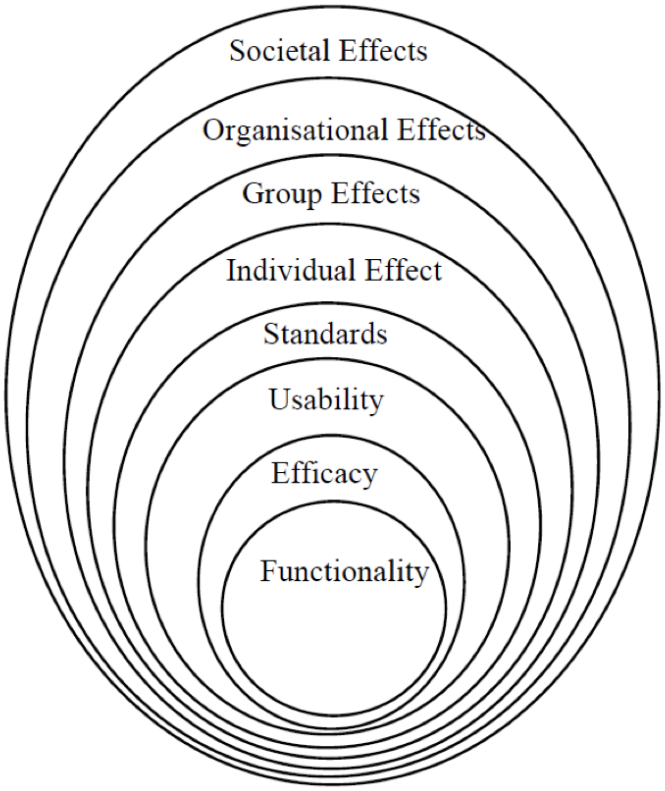


Figure 2.4: Eight layers of evaluation from SESL(Ramage, 1999).

3. Is it workable? *(usability)*
4. Does it follow the standards laid down by various bodies? *(standards)*
5. What does it do to those who work with it? *(individual effect)*
6. What does it do to their work? *(group effects)*
7. What does it do to those they work with and for? *(organizational effects)*
8. What does it do to the world beyond work? *(societal effects)*

2.2 USER INTERACTION

User interaction in crowdsourcing applications is mostly performed through user interfaces. Pan and Blevis (2011) point that one of the main crowdsourcing challenges is how to design user interfaces that can attract and sustain numerous people to join in crowdsourcing. In other words, user interfaces need to engage people as users of the system. It is necessary to understand the user motivation to participate, and designers should incorporate motivation elements to design incentive mechanisms (Zhao; Zhu, 2012).

Nielsen (2006) warns that user participation in online applications tend to be unequal: few users are responsible for contributions and most users just consume other's

contribution. So crowdsourcing applications often need to engage even more users to have a critical mass. This means that the number of people who contribute is due to the low ratio of those who are hard contributors among the total sum of users. Through user experience design it is possible to create and maintain user engagement in software (Anderson et al., 2010). User experience design covers not only building an engaging user interface, but it also has the objective of building user engagement to make users more focused on accomplishing their goals within the application (Anderson et al., 2010). Usability is a quality attribute that evaluates user interfaces (Nielsen, 2010). Through an interaction design process a designer have the means to design user interfaces to create engaging user experience with satisfactory usability (Sharp; Rogers; Preece, 2007).

2.2.1 User Engagement

It is a challenge to find ways to engage and guide the participants' attention for useful purposes (Liu et al., 2011). Anderson et al. (2010) state that user experience can drive the creation and maintaining of user engagement in software - good user experience enhances user engagement. In games, engagement is mostly seen as immersion (Anderson et al., 2010). A way to evaluate if a player is immersed into the gameplay is by verifying the lack of awareness of his surroundings (Sweetser; Wyeth, 2005). However, in non-game contexts this rule does not apply, engagement can be measured using other metrics, such as user participation.

Anderson et al. (2010) also define user engagement as the smooth accomplishment of goals. When users are using the application, to be useful, the user interface can not be an obstacle to the user path. The software must be as clear as possible and has to present the smoothest path to allow the user to accomplish his/her goals (Anderson et al., 2010).

User engagement is a special requirement for collaborative systems. Collaboration requires mutual engagement of participants mediated by some coordination in order to solve the target problem of the collaborative software Dillenbourg et al. (1995).

Trustworthiness is also important for user engagement. A good user experience gains and maintains the trust of the user about the system - trust that the software is secure and provides good content (Anderson et al., 2010). One of the key factors in the success of user engagement is user participation (Liu et al., 2011).

2.2.2 User Participation

The Inequality Participation phenomenon was highlighted by Hill et al. (1992) and disseminated by Nielsen (2006). This phenomenon occurs in online communities where 90% of users just lurk and never contribute, 9% contribute casually or with less importance, and just 1% of users are responsible for the most action in the community. Figure 2.5 illustrates the 90-9-1 proportion.

Nielsen (2006) illustrates this point with the number of Amazon book reviews. 167,113 reviews were made by only a few of the top 100 reviewers and the most prolific reviewer had written 12,423 comments. It is a classic example of Inequality Participation. Other evidence of Inequality Participation in online communities is Wikipedia where the majority of users just read the content and a small group of users edit and create content

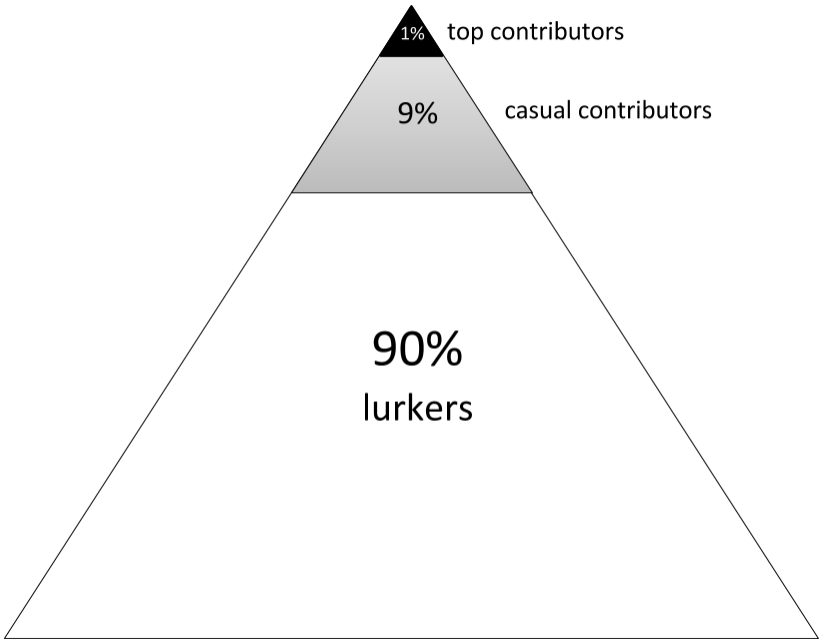


Figure 2.5: Nielsen's Inequality Participation (Nielsen, 2006) .

(Nov, 2007).

Despite this, there are ways to balance this ratio to enhance collaboration Nielsen (2006) suggests some ways: make contribution easier; make participation a side effect; edit rather than create content; wisely rewarding participants; promote quality contributors. In summary, the developer must pay attention to the user experience.

2.2.3 User Experience

In software development, the experience a user has when he/she interacts with software is called user experience (Anderson et al., 2010). User experience design combines knowledge from science, technique, craft, and art to create user experiences that effectively engage target users (Banerjee, 2004). A practical way to design user experience is through prototypes. Prototypes give the impression of the user experience and there are different kinds of prototyping that are appropriate for different kinds of evaluations (Sharp; Rogers; Preece, 2007).

The user experience of software should harmonize elements in the three levels of design: visceral, reflective, and mainly behavioral (Banerjee, 2004). The visceral level relates to how the user takes quick decisions and fast judgments which are almost automatic, while reflective design is about understanding, rational and appreciation (Norman, 2004). The behavioral level involves the brain processes that controls most of the users actions (Norman, 2004). Despite the importance of the three levels, user experience design principles are mostly concerned with how to determine how users behave (Rogers; Sharp;

Preece, 2011). We can think upon these aspects of design intuitively, but when we are able to consciously design for all levels of human cognition and emotion, then we can have a chance to create more satisfying and delightful user experiences (Banerjee, 2004).

2.2.4 Usability and Usability Tests

Nielsen (2010) defines usability as a quality attribute that evaluates how easy user interfaces are to use. The absence of frustration in using software can tell how usable that software is (Rubin; Chisnell, 2008). According to Krug (2000), the most important point to achieve a good usability is not making the user think. When users have to think about how to use a user interface, they get distracted from the task at hand (Krug, 2000). To provide better understanding, usability is often defined by: effectiveness; efficiency; safety; utility; learnability Sharp, Rogers and Preece (2007).

Effectiveness refers to how the software behaves as expected and how easily users can use it to do what they want (Rubin; Chisnell, 2008). This can be measured quantitatively through some rate of success or error - percentage from users who achieved expected results from total users. For example, effectiveness can assume the value of 80 percent of all users which were able to use a software function properly on the first usage. Efficiency is related to how quickly the user accomplishes his/her goal accurately and completely and is usually a measure of time (Rubin; Chisnell, 2008). For example, we can say that the efficiency of a function is of 80 percent of all users, because they were able to use the function properly within 30 seconds.

When users feel safe or trust the software we can say that it accomplishes safety. A way to tell that a software usable is whether users can do what they have to do without hesitation or questions (Rubin; Chisnell, 2008). Learnability is a part of effectiveness and is about how the user acquires competence to use the software (Rubin; Chisnell, 2008). In crowdsourcing applications, it is rather impossible to teach every user how the user interface works. In these scenarios, learnability is very important, and the user interface must be intuitive to let the users learn how to use it by using it.

Utility, or usefulness, concerns the degree to which a product enables users to achieve their goals (Rubin; Chisnell, 2008). Without utility other measures do not make sense. Even when some software is easy to learn how to use, it is efficient, it is effective, it is safe, if it does not achieve the specific goals of a specific user, it will not be used (Rubin; Chisnell, 2008). Usefulness is probably the most important element of usability.

Rubin and Chisnell (2008) recommends that usability must be tested in the early stages of the user experience design. In a usability test, users are shown something (an application, a prototype, sketches of an individual screen) and then are asked to either figure out what it is, or try to use it to perform a specific task (Rubin; Chisnell, 2008).

A usability test is a kind of experiment or part of one. Experiments are thought to have greater value when the users are representative. However, it is much more important to test early and often than test representative users, so the importance of recruiting only representative users is overrated (Rubin; Chisnell, 2008). In other words, we must test early, test often, and test usability.

Usability tests can make use of concepts such as Usability Heuristics for User Inter-

face Design (Nielsen, 1995) and Interface first (Fried; Hansson; Linderman, 2009) - as programming is more expensive and harder to change than user interface, it is better to design the user interface first. The point of testing is not necessarily to prove or disprove something, but to give resources to make the designer judge the usability (Rubin; Chisnell, 2008).

There are several kinds of usability tests, such as Five-Second Tests, Preference Tests, Click Tests, Navigation Flow Tests. The Five Second Test works to check first impressions of a screen and how easy the screen is understood. Preference tests show two versions of the same studied artifact and ask what the tester prefers according to a given criteria. Click Tests and Navigation Flow Tests evaluate the tester behavior through their interaction on the screen. These kinds of tests are valuable to scenario where there are the original screens and the screens generated after the treatment under evaluation.

UsabilityHub[1] is a tool used to run usability tests by providing the creation of private and public tests. Private tests must be assigned to known testers. Public tests are assigned to at least 10 community testers. Nielsen (2000) states that we can evaluate the usability with up to 5 users running many small tests and still get the best results.

The Usability Hub has a karma-based economy as shown in Figure 2.6. For every usability test taken on UsabilityHub, the community tester is rewarded with 1 Karma point. To run your own usability test, it costs at least 10 Karma points. Therefore, the main motivation for a community tester in taking a test is to gain karma points so that they can create their own usability tests.

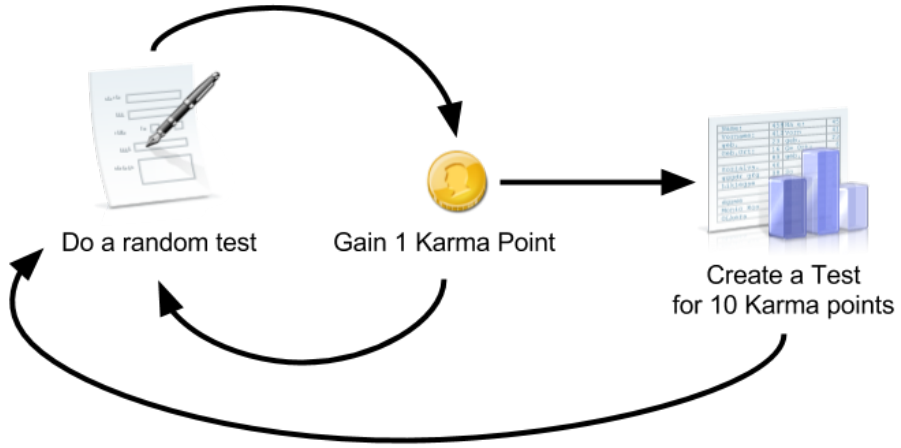


Figure 2.6: UsabilityHub Karma Points.

Public test responses come from UsabilityHub members (Group A), as well as the people that Group A invited to complete tests, the Group B (Usabilityhub, 2015). Group A is made up of designers, developers, UX experts, and website owners who probably are conducting usability tests on UsabilityHub (Usabilityhub, 2015). Group B is made of friends and colleagues of Group A who are casually taking tests on UsabilityHub. According to UsabilityHub, the proportion of users in both groups is about a 50/50 split (Usabilityhub, 2015).

[1]http://www.usabilityhub.com

2.2.5 Interaction Design

Interaction design can be defined as the practice of designing interactive digital products (Banerjee, 2004). An objective of interaction design is to support people in their daily activities (Sharp; Rogers; Preece, 2007). Interaction design articulates user experience goals which cover a range of emotions, including desirable and undesirable ones (Banerjee, 2004). Examples of desirable aspects are satisfaction, enjoyment and engagement while undesirable aspects refers to boredom, frustration and guilt (Banerjee, 2004).

Interaction design focuses more on designing the behavior and how it relates to form and content (Banerjee, 2004). But when a user behaves on a user interfaces, he/she expects some feedback. User interfaces should be designed to provide proper feedback to the users to ensure they know what they are doing and what to do next (Banerjee, 2004). Sharp, Rogers and Preece (2007) presents the interaction design as a process of 4 main stages:

1. Identifying needs and establishing requirements;
2. Developing designs that meet established requirements;
3. Building interactive designs;
4. Evaluating interactive designs.

The first one can be performed by studying the scenario status quo and by requirement analysis. Then developing designs can be drawn as sketchy prototypes, and building interactive designs (functional or non-functional). Then usability tests can be performed to evaluate these prototypes.

2.2.5.1 User Stories User stories are a simple and effective way to express software requirements in a language that developers and non-developers can understand (Wake, 2003). A user story usually has the form:

- *As a **[user]**, I want **[function]** so that **[value]**.*

In one sentence we describe the 3Ws of a software requirement: the who; the what; and the why. An example of a user story for an application where a reviewer can write comments on a blog post could be written in the form:

- *As a **reviewer**, I want to **post a comment** so that I can **express my opinion about a post**.*

Writing the user story in the first person makes the software developer imagine the software requirement from the user perspective. Another good point about user stories is the value part. By forcing the developer to define the value, he/she reviews the purpose of a certain function and why it must be accessed for that user.

2.3 GAMIFICATION

Digital marketing and social media professionals use game design techniques to engage consumers in their target products, this approach is called gamification (Liu et al., 2011). In 2010, gamification became a trend and interest continues to according to Google Trends [2]. McGonigal (2011) argues that if games engage players to solve fictitious problems in a virtual world, these same players could act together to solve real world problems. The idea of using game design techniques to encourage user participation in different contexts has been increasingly studied in several contexts (Raymer, 2012; Mason; Michalakidis; Krause, 2013; Lund; Coulton; Wilson, 2011; Burke; Marlow; Lento, 2009; Janzik; Herstatt, 2008). Gamification can also be used to engage users in applications which require collaboration.

Zichermann and Cunningham (2011) define gamification as *"the process of game-thinking and its mechanics to engage users and solve problems"*. The authors believe that dealing with gamification this way could help solve problems influencing human motivation and behavior. By game mechanics, the authors Zichermann and Cunningham (2011) consider tools that may result in meaningful response from the players, for instance: points, levels, leaderboards, badges, challenges/quests, onboarding and engagement loops. Game-thinking sounds interesting to engage users in problem-solving, however, how the authors present the definition is more about psychology concepts for human motivation than practical techniques that can have measurable results.

Huotari and Hamari (2012) define gamification as *"a process of enhancing a service with affordances for gameful experiences in order to support user's overall value creation"*. This definition emphasizes the goal of gamification: richer experiences rather than the methods themselves. This definition maybe be better for commercials purposes, mainly marketing as a service, where the product value is more important, but collaboration depends also on users relationships, community-thinking and team-work. So, it is about how people work together to build and maintain the value of a service.

Deterding et al. (2011) defines gamification as the *"use of game design elements in non-game contexts"*. It is about take advantage of theory and practice of well-known concepts to design games in other contexts that are not games. From this, we can use game design techniques in collaborative software. This definition for gamification will be used as the main gamification definition.

Since Gamification is a recent field, there is no consolidated evaluation method. Therefore, after researching evaluation on game development literature, we thought it suitable to judge gamified applications using some criteria used to evaluate serious game. Serious games differ from ordinary games because they have a non-entertainment primary purpose (Michael; Chen, 2005). In our scenario the main purpose is collaboration. Although a gamified application is not a game, even a serious game, we could use serious game evaluation criteria to evaluate the game aspects of a gamified application. The following criteria can be used, as defined by (Michael; Chen, 2005):

1. Active involvement and stimulation of all players;

[2] http://www.google.com/trends/explore#q=gamification - Google trends for "gamification"

2. Sufficient realism to convey the essential truths of the simulation;

3. Clarity of consequences and their causes in both rules and gameplay;

4. Repeatability and reliability of the entire process.

The first criterion concerns the active involvement and stimulation of players. The second criterion is how the game provides sufficient realism to convey the essential truths of the simulation. The third criterion is the clarity of consequences and their causes in both rules and gameplay. For this, we must evaluate mostly the feedback system. The player must have an awareness of what their action will result in. This result must be compared to verify if it was as expected. The last criterion is about the repeatability and reliability of the entire process. We must evaluate the whole player interaction loop.

2.3.1 Game design

Games have successful strategies to grant enjoyable user experience (Deterding, 2012). McGonigal (2011) states that games provide unnecessary obstacles that a player volunteers to overcome because they provide positive stress, emotion, meaning and feelings of accomplishment. Studies shown that playing games stimulates pleasure centers in the brain making people enjoy games (Xu, 2011). According to Schell (2008), we are in a new era where design is about what is pleasurable instead of what is efficient.

Game Theory studies scenarios of strategic decisions where the gain of each participant depends on the others (Pimentel et al., 2011). Each participant decides his/her strategy after assessing the situation of other participants and makes assumptions about the strategies they adopt (Pimentel et al., 2011). Based on this theory, we illustrate the strategic decision process of participation in Figure 2.7. The game consists of making a decision on what action to perform, perform the action and receive a result for the action, then the player decides to do a new action or not (Pimentel et al., 2011).

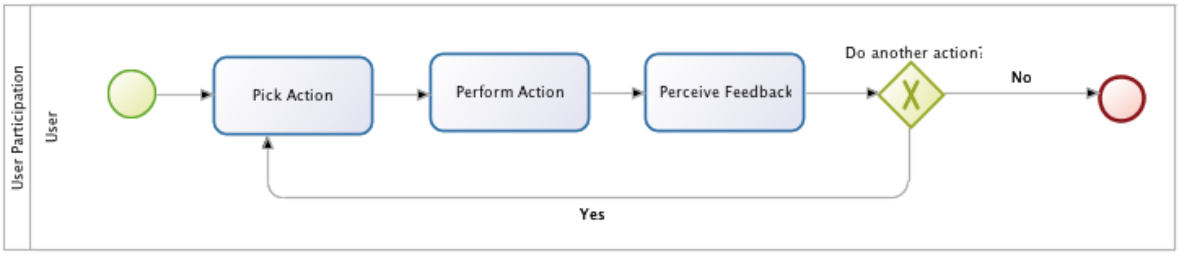


Figure 2.7: Game Loop (Schell, 2008).

In Game Theory, the Normal form is a representation of every player feedback for every state in the environment, where the situation of the environment depends only on the players actions combined (Leyton-brown; Shoham, 2008).

As a result, using the Normal Form representation, a game is defined as a *tuple (P, A, u), where:*

- *P is a finite set of n players, indexed by i;*
- $A = A_1, x \ldots x, A_n$, *where* A_i *is a finite set of* actions *available to player i*;
- $u = (o_1, ..., o_i)$ *where* o_i *is an feedback* function *for the player i.*

When applying these concepts in real situations such as in online communities, we can evaluate, propose and act in group scenarios for inserting incentives that foster collaboration (Pimentel et al., 2011). Even taking into account its limitations, Game Theory can provide tools to promote our comprehension and analysis of real situations involving decision making (Pimentel et al., 2011).

Games are also made up of game elements and game states. Game elements are the physical and logical components that the Player can control in a certain game state (Björk; Holopainen, 2005). Players influence the game state by performing actions on game elements (Björk; Holopainen, 2005).

According to Schell (2008), information flows in a loop from player to game to player to game - this is the loop of interaction. Figure 2.8 illustrates the Loop of Interaction. Every cycle completion generates a new experience and the returned feedback affects what the player will do next. Feedback influences how the player understands and enjoys what is going on the game.

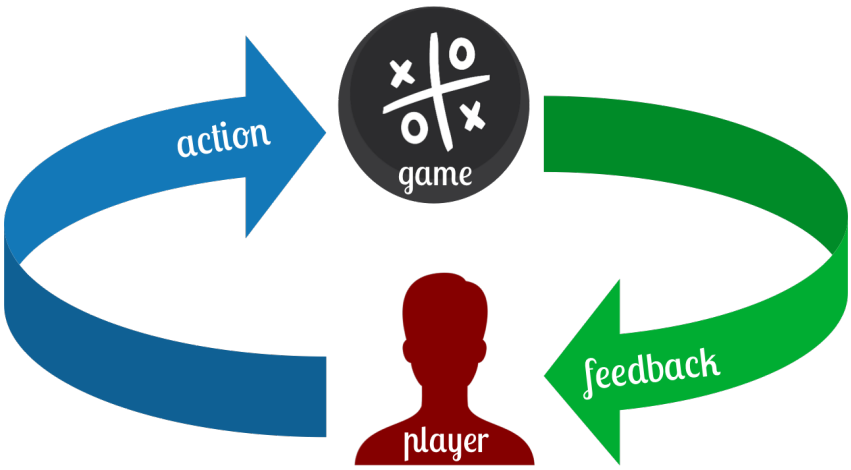


Figure 2.8: The Loop of interaction, adapted from (Schell, 2008).

Game design consists of the definition of all the concepts and behavior of a game. Schell (2008) argues that the main skill in designing games is listening, and he divides listening into five kinds: the team; the audience; the game; the client; and himself. A long time can be spent listening to information and data to design it but this can delay finding a solution (Cross, 2011). The designer should therefore not spend too much time in this phase, although it brings us awareness of the application scenario.

2.3.2 Game design patterns

Game design is the act of deciding what the game should be (Schell, 2008). It is about building all the aspects that define a game, whatever the game definition is. It is not about programming nor about graphic design, but about taking decisions about the game (Schell, 2008). A game designer can make use of some tools to achieve his/her goal, such as game design patterns.

A way for developers to solve problems is the use of patterns and game design is no different. A pattern describes a problem that often occurs in a certain environment, and concerns the core of the solution to that problem in a way that it is possible to replicate the use of the solution (Björk; Holopainen, 2005). Game design patterns refer to prototypical solutions that can be implemented with many interface elements (Deterding et al., 2011). Björk and Holopainen (2005) organizes in 11 categories, 296 game design patterns for:

1. Game Elements;
2. Resource and Resource Management;
3. Information, Communication, and Presentation;
4. Actions and Events;
5. Narrative Structures, Predictability, and Immersion Patterns;
6. Social Interaction;
7. Goals;
8. Goal Structures;
9. Game Sessions;
10. Game Mastery and Balancing;
11. Meta Games, Replayability, and Learning Curves.

The searching for patterns in (Björk; Holopainen, 2005) can start from the categories, and then explore its patterns. For example, design patterns related to collaboration, looking within the game design patterns for Social Interaction, we find the Cooperation pattern. This pattern happens *when players coordinate their actions and share resources, in order to reach goals or subgoals of the game* (Björk; Holopainen, 2005) and some ways to achieve it is to use patterns such as Team Play or Mutual Goals. A pattern can be related to others by instantiation, modulation and conflicts. Björk and Holopainen (2005) describes the consequences of the patterns, for example, the Cooperation pattern increases Social Interaction (another pattern) between players. So, patterns have relationships with other patterns. Another example is if Cooperation and Competition patterns coexist in the same game, it also increase the Tension between players. Table 2.1 showss how Björk and Holopainen (2005) describe the Cooperation pattern.

Name	Cooperation.
Description	*Players cooperate, i.e., coordinate their actions and share resources, in order to reach goals or subgoals of the game.*
Using the pattern	*The simplest way to achieve Cooperation is to use Team Play, but equal levels of Cooperation can be achieved by introducing Mutual Goals with Shared Rewards to players.*
Consequences	*Cooperation increases Social Interaction between the players, as they have to coordinate their actions in order to reach the goals of the game.*
Example	*Games where there is no possibility for destructive player versus player actions, such as attacking or stealing, encourage Cooperation as the possibility of Betrayal is lessened. Further, a player that does not cooperate can lose compared to the other players if all the other players collaborate.*
Instantiates	*Constructive Play, Alliances, Betrayal, Social Interaction.*
Modulates	*Competition, Tension, Dynamic Alliances, Team Play.*
Instantiated by	*Collaborative Actions, Shared Rewards, Team Play, Mutual Goals.*
Modulated by	*Social Statuses, Competence Areas, Shared Resources, Bidding, Trading, Delayed Reciprocity, Individual Rewards, Social Dilemmas, Social Organizations.*
Potentially conflicting with	*Conflict, Betrayal.*

Table 2.1: Cooperation Game Design Pattern (Björk; Holopainen, 2005).

2.3.3 Gamification approaches

Most gamification design guidelines (e.g., (Dubois, 2012; Liu et al., 2011; Zichermann; Cunningham, 2011; Duggan; Shoup, 2013; Nicholson, 2012)) deal with introducing game elements based on points, badges and leaderboards. Liu et al. (2011) present the cycle of gamification as an environment surrounding a points system, taking steps: challenges, victory conditions, rewards, rankings, medals, status and social networks. The scoring system as the core of the system can result in gamification score as the primary purpose of the application. This approach can lead the users to carry out the activities merely in order to compete and / or in order to achieve individual goals rather than participate collaboratively in order to achieve the collective goals.

On the other hand, the method presented by Dubois (2012) is inspired by two concepts: self-organizing systems and the existing experience in game design. Through

classes of self-organization (collaborative, competitive and environmental), four requirements are raised for the application: user motivation, context-sensitive learning, reward system, and collaborative creativity. This method focus on self-organizing systems and do not address applications requiring mediation actions. Another feature is that the guidelines presented by the methodology serve as an aid only to the design stage of an application, specifying guidelines for architecture, implementation and maintenance of the application.

Another contribution is the Gamification Design Process by Werbach and Hunter (2012). This work presents 6 steps to gamification: 1) define business objectives; 2) delineate target behaviors; 3) describe yours players; 4) devise activity cycles; 5) don't forget the fun; and 6) deploy the appropriate tools.

Gamification is not about turning applications into games, however, we can find some similarities between games and collaborative software - which makes the use of gamification in collaborative software appropriate. All games have four basic characteristics: goal; rules; feedback system and voluntary participation (Mcgonigal, 2011). The goal is feedback that players pursue, while rules place limitations on how players can achieve their goals. The feedback system shows the player how close or distant he/she is to achieving the goal, and voluntary participation shows that each player knows and voluntarily accepts the goal, rules and feedback. Collaborative software has similar traits.

Collaborative software seeks to support groups of people to reach a common goal (Ellis; Gibbs; Rein, 1991). A good collaborative purpose which provides a shared outcome among the group drives voluntary participation (Potter; Mcclure; Sellers, 2010). Users need consistent and constant feedback about what is happening (Seltzer; Mahmoudi, 2012) and positive feedback can enhance the intrinsic motivation of the participants(Janzik; Herstatt, 2008), i.e., when users receive feedback on their contribution, they increase their participation (Duy; Hoang; Wang, 2010). Finally, the collaboration also limits how the user will use the system, but (unlike games) there are also limitations provided by the real world.

2.4 GAMIFICATION IN CROWDSOURCING

This Section discusses how Gamification is used in collaborative software both in commercial solutions as in the technical literature. There are several commercial initiatives of gamified crowdsourcing, or crowdsourcing software with gamified features. An example is Waze[3], a community-based traffic and navigation app, where drivers share real-time traffic and road information. The goal of Waze is to enable users to take the best route in their daily commute. Drivers anonymously share information about the lane speed based on their drive speed. As Figure 2.9a shows, users can create and receive alerts about location of police, accidents, road hazard and traffic jams. When the user is near a reported alert, Waze pop-ups a dialog so the user can confirm that the object reported is not there any longer as shown in Figure 2.9b. There are other features such as sharing position with friends, coordinating friends for a pick-up and finding best gasoline prices nearby.

[3]http://waze.com - Community-based traffic and navigation app

(a) Waze dialog to create alerts . (b) Waze dialog to confirm an alert.

Figure 2.9: Waze interfaces for alerts.

Another example of gamified collaborative software is Foursquare[4], a local discovery mobile service. Basically, a user tells Foursquare the place he/she is with check-ins and then in response, it provides information about the corresponding place. Based on things the service learns about the user's preferences, Foursquare suggests places to go. As recommendations are based on user behavior, it is very important that the user provides as much information as possible. Gamification elements in Foursquare, based on badges and achivements, encourage people to check-in often and provide Foursquare with more data that can be used to enrich the user experience (Frith, 2013).

Stackoverflow[5] is a Q&A (Questions and Answers) network where members post questions about software development, and users cooperate with each other by answering questions. To encourage contributions, StackOverflow employs gamification in the following way: questions and answers are voted on, and the votes reflect the user's reputation and badges, besides this, recruiters use Stackoverflow user reputation as a measure of the developer (Vasilescu; Filkov; Serebrenik, 2013). In this way, developers are stimulated to ask relevant questions and to give high value answers.

[4]http://www.foursquare.com
[5]http://stackoverflow.com

2.4.1 Gamification solutions in the literature: Mapping study

In the scope of this dissertation, during the Collaborative Systems course at this master program, four colleagues and I developed a systematic literature review that investigated how gamification is applied in collaborative software. The results of this study were reported in Bastos et al. (2013). In this section, we describe the defined protocol and discuss the results found.

Research Questions

The general question was "What is the state of art of gamification applied in collaborative systems". This question also looks for answers to the following questions:

1. Which frameworks (processes) and platforms (tools) have been used to implement gamification in collaborative systems?
2. Gamification has been used for what purposes?
3. What are the benefits and limitations of gamification in collaborative software?

Search strategy

The search was performed in 3 electronic online databases: ACM Digital Library[6]; IEEE Xplore[7]; and Springer Link[8]. These databases were selected because they manage papers from the most relevant conferences and journals in the area of collaborative systems. After 18 trials refining the search string which varied from 60 to 13248 results, the following search string was used: (gamification OR gamified OR "game elements" OR "game element" OR "game mechanics" OR "game mechanic") AND ("collaborative system" OR "collaborative software" OR collaboration).

Inclusion and Exclusion Criteria

This study considered only primary studies in the format of peer-reviewed paper written in English or Portuguese that reported practical experiences with gamification. The study date was not considered as an exclusion criteria.

Study Selection

The studies selection was divided into three stages: 1) search on the databases; 2) applying exclusion criteria after reading title and abstract; and 3) full papers reading. 961 papers were found in the first stage, and then this was reduced to 57 papers after the

[6]http://dl.acm.org
[7]http://ieeexplore.ieee.org
[8]http//link.springer.com

exclusion stage. After full reading, the selection was further reduced to 18 papers. The selection was made by a double review, where two different investigators reviewed each paper, and any conflict resolution was carried out by a third investigator. Although gamification is a field of interest for many professionals, there are relatively few studies exploring its use in collaborative software.

Data extraction

The following data were extracted from the studies: publication year; publication field; framework or method used; platform or tool used; benefits; and limitations. From the publication year we found an increasing number of papers reporting gamification in collaborative systems.

Gamification was mostly used in tagging applications and social networks. There were also applications in education; citizen science; collaborative editing; information delivery and visualization; discussion groups; democratic meetings; process modelling; BPO Industry; and Question & Answer.

Most applications were implemented without a supporting framework, meaning, they were ad-hoc implementations. The ones that used some supporting framework, e.g. Drupal [9] and Hadoop [10], used general purpose frameworks, i.e. not specific gamification frameworks.

The identified benefits of gamification in collaborative software were: user engagement; fun; social and emotional impact. Most papers did not present any limitations, however, those that did, mentioned that it was: hard to arrange different motivational factors; not relevant impact on user behavior; and hard to balance the point system.

Analysis of the results

The study indicated a lack of consolidated approaches to support the implementation of gamification in collaborative software. No method or conceptual framework to guide developers within conceptualization, planning, and design of gamification on collaborative software was found. Moreover, the literature lacks frameworks or platforms suitable for implementing gamification in collaborative software.

2.5 CONCLUDING REMARKS

User interfaces of crowdsourcing applications need to encourage users to participate through user experience that engages users to be more focused on the accomplishment of their goals within the application. Usability is a quality attribute that evaluates user interfaces (Nielsen, 2010). Through an interactive design process a designer have the means to design user interfaces that create engaging user experience with satisfactory usability, and this process can be defined in four main stages: requirements analysis;

[9] http://www.drupal.org - Open Source Content Management Platform

[10] http://hadoop.apache.org - Apache Hadoop is a framework that allows for the distributed processing of large data sets across clusters of computers

interface design; development or prototyping; and evaluation.

User interface design is a challenge to attract and sustain numerous people in crowdsourcing Pan and Blevis (2011). And the first aspects to evaluate on a collaborative system are its functionalities, efficacy, usability and the use of standards (Ramage, 1999). So, what the software provides of functionalities, whether they work well enough, how users can interact with it and if it follows the well-known standards

Gamification has been widely used to engage user and motivate them to participate. The principle is based on the fact that if games engage players to solve fictitious problems of a virtual world, these same players could act together to solve real world problems (Mcgonigal, 2011). Despite the increasing interest in gamification for many reasons (Deterding, 2012), most current gamified applications will fail to meet business objectives primarily due to poor design (Gartner, 2012). This scenario suggests that it is not clear how gamification works and how to implement successful gamification in online applications.

Game design elements have been used in collaborative software to engage users in the collaborative goals. There are several examples of gamified user experience to foster collaboration such as Foursquare, StackOverflow and Dropbox. Most gamification methods are too general and do not emphasize collaboration aspects. Nevertheless, these methods only make use of superficial game design elements (e.g., points, badges and leaderboards). Reward-based which is dangerous in collaborative systems because it may result in making points as the primary purpose of the application instead of collaboration. This approach can lead users to carry out the activities merely to compete and/or to achieve individual goals rather than participate collaboratively to achieve the collective goals.

So far, we have not found any specification of how collaborative software can be organized to promote the application of gamification techniques in the design, development and evolution stages. A systematic user participation design process can be used not just to design the application in early stages but also to constantly manage user activities to fulfill participation requirements.

Chapter 3

THE G.A.M.E. CONCEPTUAL FRAMEWORK

In this chapter we introduce G.A.M.E. (an acronym for Gathering, Analysis, Modeling and Execution), a conceptual framework that aims to support developers in designing gamification for user interaction to increase collaboration in crowdsourcing systems. The framework provides a process that combines knowledge from software engineering, collaborative software, game design and interaction design. The overall structure of G.A.M.E. can be seen as a specialization of the waterfall model used in software development processes with particularities of gamification applied to collaborative software.

G.A.M.E. was built following a bottom-up and top-down approach, using as reference two collaborative systems from the domain of public transportation systems, developed in the scope of the Ubibus Project [1] (Vieira et al., 2012b). The first system, Ubibus-cars (Vieira et al., 2012a) was used in the bottom-up approach to support identifying the concepts. The second system, Wikibus (Moitinho, 2013) was used in the top-down approach to validate the defined concepts, and it is presented in Section 4.1.

From our experience in gamifying crowdsourcing systems we propose a framework to guide the design of gamification in collaboration, so developers can get better results than developing it in an ad-hoc way. Using a systematic approach, studies can be replicated and compared, which is an important advance for gamification and collaborative fields.

In summary, our approach is organized in four phases: gathering, analysis, model and execution. Figure 3.1 shows the G.A.M.E. using BPMN[2], where each phase is represented as a sub-process. In the first stage, we gather all useful information about the target application regarding its goals, technology, functionalities and issues. After that, we analyze the data gathered from the collaboration perspective. Then we model the gameplay using game design techniques. Finally, we execute tasks regarding the development and testing of the gamified application. After that, the whole process can be repeated. We created a gamification spreadsheet template [3] to help the designer to write the outputs of each step

[1] http://www.cin.ufpe.br/ ubibus - The UbiBus Project, Ubiquitous and Context Sensitive Public Transportation System.

[2] Business Process Model and Notation - http://www.omg.org/spec/BPMN

[3] https://goo.gl/5SQutM - G.A.M.E. spreadsheet template

and use them as inputs for further steps. This spreadsheet also provides to the designer a sense of progression during the gamification process. We expect also that it serves to compare G.A.M.E. instantiations. The overall purpose of this framework is to support turning an ordinary user experience into a gameful experience - with qualities of gaming. In the following subsections, each phase will be covered and detailed.

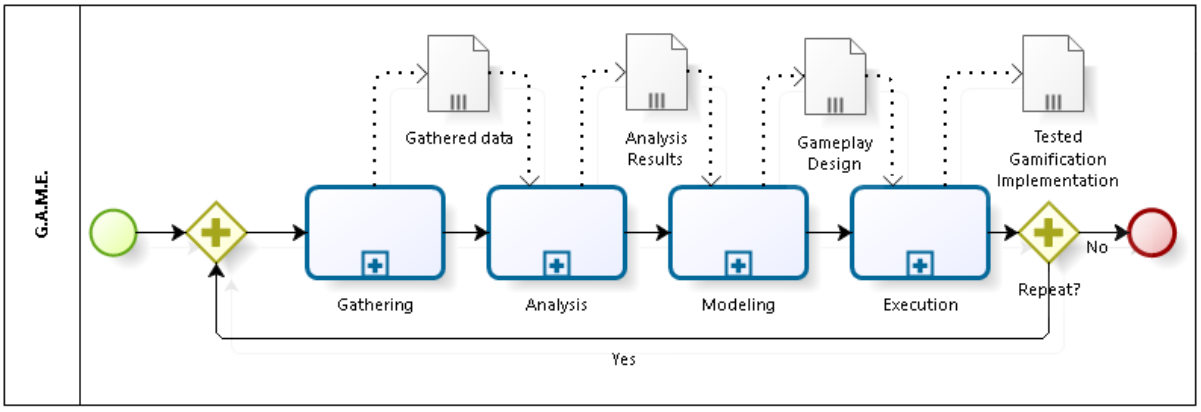


Figure 3.1: G.A.M.E. Process.

3.1 GATHERING: UNDERSTANDING THE APPLICATION BACKGROUND

As discussed in Chapter 2, one of the main skills in designing games is listening and this is the starting point which is explored in this first phase. The main goal of this phase is to understand the application scenario by collecting information about the application goal, technology, functionalities and issues as illustrated in Figure 3.2. Listening to the game means knowing it from the deepest elements to the most visible to the user. As we now see the application as a game, the same rules apply. To listen a collaborative software, we first argue what is its goal, and then it derives other participants to listen to.

The sequence proposed for the gathering is intended to drive the developer to obtain information that will be useful in the following steps. Figure 3.3 illustrates how these 4 aspects relate to each other and to collaboration. The goal is what the participants hope that the community will achieve and can be accomplished by the software functionalities. The software is supported by technological infra-structure that we represent as the technology block. The technology implements functionalities that enable collaboration. The collaborative software may have issues that could limit the collaboration, which might disturb the goal and break some functionality.

We can also notice that we can influence the technology, the functionalities and the issues. The goal itself hardly changes, because it mostly defines the collaborative software. We can change the technology if it does not implement well the expected functionalities. We can update the functionality to better reach the goal. We can solve issues to stop limiting or breaking functionalities and disturbing the goals.

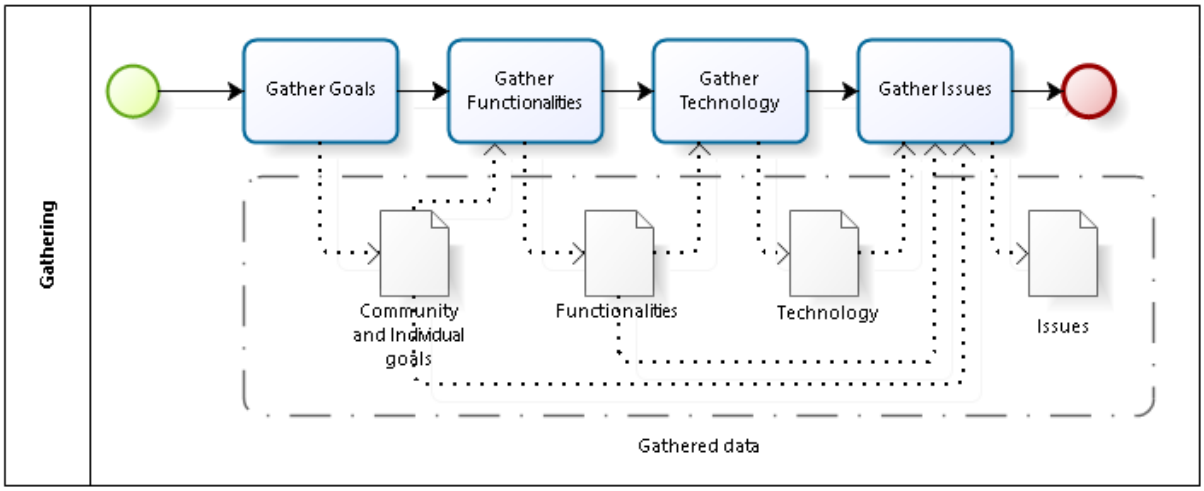


Figure 3.2: Gathering sub-process.

3.1.1 Gather Goals

As discussed in Section 2.3, a goal in game design is what the players hope to achieve by playing the game, and it provides a purposefulness for the players. It serves as incentives by focusing players' attention and continuously stimulating their participation. However, in collaborative software, incentives not only depend on the motivation of the users, but they also have great relevance to the community purpose.

Motivation to participate occurs at two levels, the individual and the community, and while individual and community interests in collaboration may be aligned, they remain separated (Potter; Mcclure; Sellers, 2010). In our framework, we need both concepts: the individual and the community goals. If the player's goal is what they hope to achieve by playing the game, and in our scenario the game is a collaborative software, the player's goal must also be aligned with the collaboration. For example, in the Waze system (WAZE MOBILE LTD., 2009-2015) the community goal is to help drivers to have a better traffic experience by providing them the best route with real-time information. Examples of individual goals of Waze users include: getting the fastest or the shortest route; notify or get notified about incidents on their routes; share location information with friends during commuting. Another example is within Ubibus-Cars, where community goal is to support and encourage individuals who perform a common route to use the same vehicle, thus supporting the public transport system. And an ideal individual goal could be to share the vehicle to reduce the traffic.

In summary, the collaborative software goal is not only the individual goal, but also the community goal. The ideal individual goal is to participate in a way that help the community to achieve its goal. When the developer is aware of what the collaborative software goals are, it becomes easier to identify how the functionalities that provide means to achieve them work.

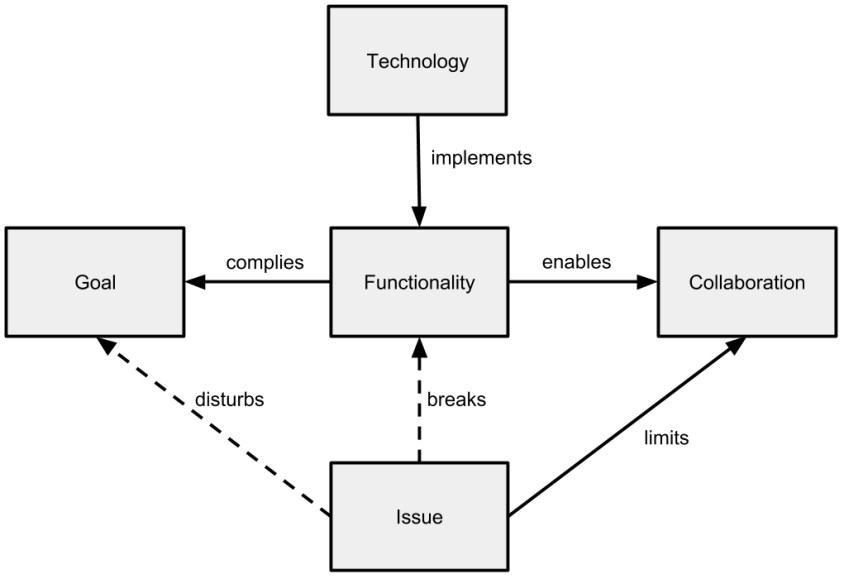


Figure 3.3: Relationships of the gathered data.

3.1.2 Gather Functionalities

This activity aims to identify the functionalities that allows the user to interact with the collaborative software. It must comply with the community and the individual goals. For example, Waze functionalities are related to traffic information and navigation. Its main functionalities are: routes searching and navigation; viewing and creating transit incidents; and sharing location with friends. Ubibus-cars goals are basically: ride offer and requests management; review of rides (drivers and passengers as well).

3.1.3 Gather Technology

Collaborative software uses technology to implement functionalities that attend the application goal and enable collaborative aspects to arise. For example, a collaborative software that runs on a certain device and it communicates with other devices directly or through a network, the collaboration objective may be affected by the network reliability.

It is also important to know how the software architecture is implemented. The kind of device, programming language or libraries used can place rules on how it complies with the goal and enables collaboration. In our approach, we suggest the designers to gather information regarding: platform; operational system; architecture; libraries or frameworks.

3.1.4 Gather Issues

Issues are problem that limits the collaboration in the collaborative software. Collaborative software such as Wikibus (Moitinho, 2013) and Ubibus-Cars (Vieira et al., 2012a) may face challenges from the technology employed for human usage. For example, in Waze where the users are the main providers of information, if the incidents notifications

are not true or the suggested routes are not consistent, users could stop using it, and both individuals and community goals would be affected. Issues related to software malfunctioning can break functionalities and impact the goal indirectly. Issues can be discovered directly by the developer when using it or by observing the usage by a real user.

Previous evaluations of Ubibus-cars raised some issues that serve as example. Participants of the evaluation raised some concerns that include: ride-sharing is based on mutual trust between individuals; lack of trust on share rides with strangers; lack of trust on rides scheduling; and subjects were not aware of what happened with their requests (poor feedback). In summary, issues were related to poor usability and trust.

3.2 ANALYSIS: CROWDSOURCING AND COLLABORATION ASPECTS

The main purpose of the Analysis phase is to review crowdsourcing and collaboration aspects of the application. This phase uses the results of the study made in the Gathering phase as input. Despite the chance of making a wrong choice, we can increase the chances of succeeding by studying the data collected so far. Figure 3.4 shows the steps in the Analysis phase. The first step is the crowdsourcing analysis to have a clear view of what is the crowdsourcing target goal and what the users do to achieve it. Then, the second step will cover how the collaboration happens by categorizing the functionalities in collaboration. Finally, the last step covers the relationships between the issues with each collaborative dimension.

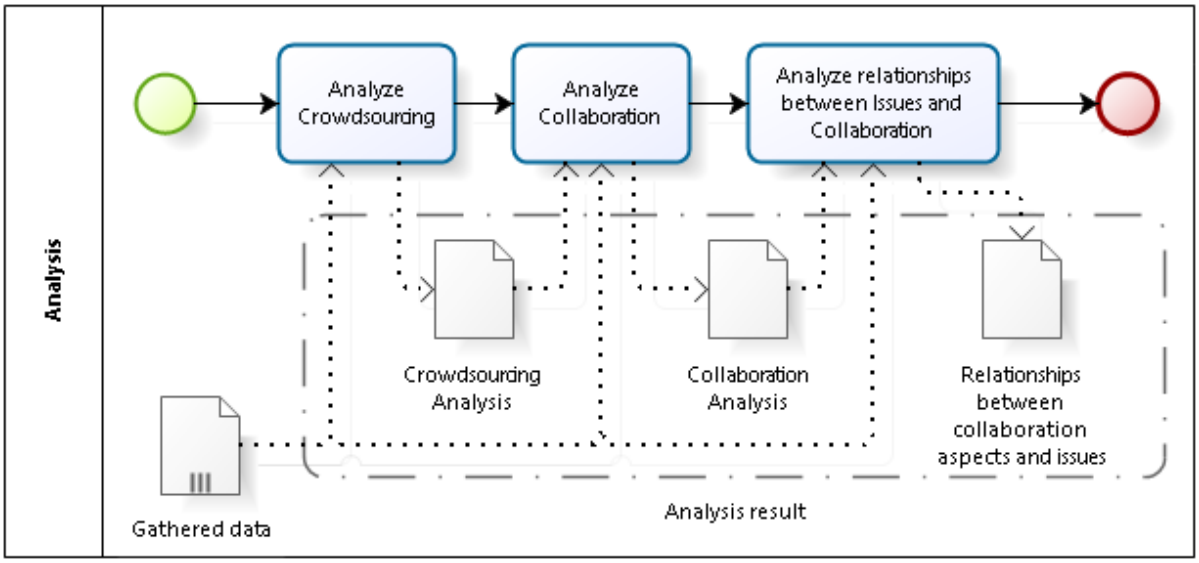


Figure 3.4: Analysis sub-process.

3.2.1 Analyze Crowdsourcing

This step analyzes the application from the point of view of crowdsourcing. In this sense, G.A.M.E. proposes to use crowdsourcing concepts as discussed in Section 2.1.2. From

the crowdsourcing classification presented by Doan, Ramakrishnan and Halevy (2011) we extract the following questions:

1. What is the target problem of crowdsourcing?
2. What do the users do in crowdsourcing?

These questions help to focus on the core of crowdsourcing. Crowdsourcing applications usually have many functionalities, and after the Gathering phase, many functionalities may have been listed, but just some of them are directly related to the target problem, and to what users do. For example, in Waze the target problem of crowdsourcing is to obtain traffic information such as incidents on lanes from drivers that pass by that lane. In this case, what the users do: post a new incident; confirm an incident posted; inform if an incident no longer exists; comment about the incident. As another example, Ubibus-cars crowdsourcing target problem is to build a social network for ridesharing, while drivers create network with other locals helping to reduce the vehicles traffic.

3.2.2 Analyze Collaboration

With the result of analysis of the crowdsourcing the developer is aware of what is the target problem of crowdsourcing and what users do to help the system to achieve it. At this point, we can split it in three main concepts of collaboration: cooperation, coordination and communication. For example, in Waze, users cooperate by posting traffic incidents, while the coomunication between users about that incident is performed through comments. We can say that the users help the coordination by confirming an incident or informing that it no longer exists. As another example, a Ubibus-cars cooperation task is to post ride offers, then to communicate users negotiate rides, and finally the coordination will consist on the review of the rides.

3.2.3 Analyze relationships between Issues and Collaboration

G.A.M.E. considers two approaches to choose a target: to enhance a strength or to fix a weakness in the application. For the first approach, we can detect a strength from the functionalities that can be converted into more collaboration results. For example, an application in which the voting system results in more (or better) collaboration than other functionalities. It is relevant to analyze how it can be even more attractive to the users. However, to fix a weakness, it is necessary to study application issues.

To understand issues, we may have to discover their possible causes. For example if its caused by a limitation of the technology used, we should change or improve the technology. Another cause could be a failing functionality, but not from the perspective of the implementation, but from the perspective of the user. If for the user it just does not work as it is, maybe game design techniques can help solving this issue.

3.3 MODELING: GAMEPLAY DESIGN

The Modeling phase (Figure 3.5) consists of identifying desired game design patterns that can support the collaboration and the writing of player stories for further implementation.

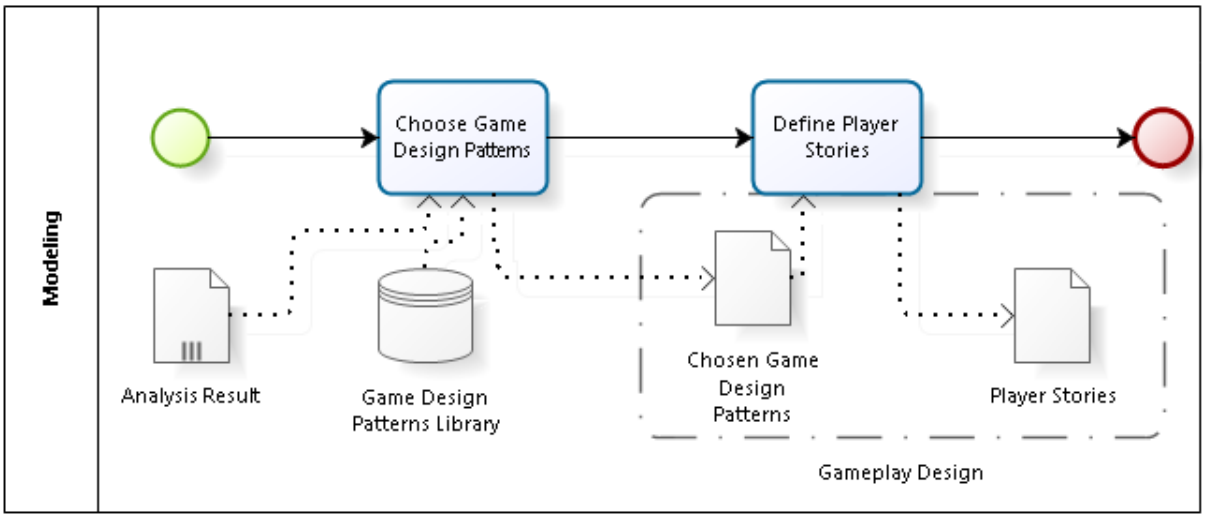


Figure 3.5: Modeling sub-process.

3.3.1 Choose Game Design Patterns

In this step, de designer should investigate desirable game design patterns that can support each collaborative task to solve the crowdsourcing target problem. We suggest the use of game design patterns described in (Björk; Holopainen, 2005) which organizes 296 game design patterns in 11 categories. Then the patterns can be studied until one is found that fits for the player story. After this, the gameplay must be designed in order to implement the game pattern.

3.3.2 Define Player Stories

As discussed in Section 2.3, Game Theory represents a game in the Normal Form as a (P, A, u), where P stands for the players, A for the actions, and u for the outcomes (Leyton-brown; Shoham, 2008). We also discussed that a player interacts with the game that results in feedback for the player and it generates a loop of interaction (Schell, 2008). Finally, games also are made up of game elements that the player can control in a certain game state (Björk; Holopainen, 2005). G.A.M.E. combines these three concepts to represent the Game Interaction Model illustrated in Figure 3.6. In summary, a player does an action in a certain game state which provides feedback to the player. In order to start the design of a gameplay we identify these elements in the target user story.

In crowdsourcing systems, players can be categorized according to their participation, experience and permission level (Figure 3.7). Their participation could be: heavy contributor; casual contributor; lurker (Nielsen, 2006). An example of experience categorization is to organize them as newcomer or veteran. And their permission level could be categorized as: anonymous; normal user; moderator; admin.

Every element of a user story can be associated with an element in the Modeling: the user in the user story is the player; the function is the action; the value is the feedback. However, the state there is no respective element in the model. In this approach, the

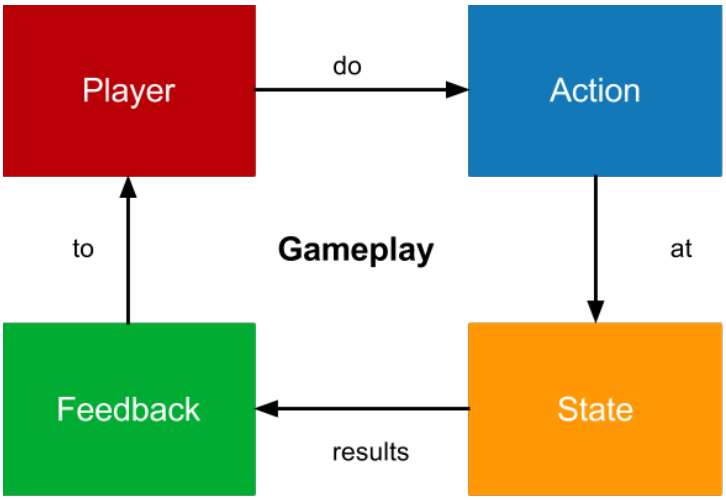


Figure 3.6: Gameplay Interaction Model.

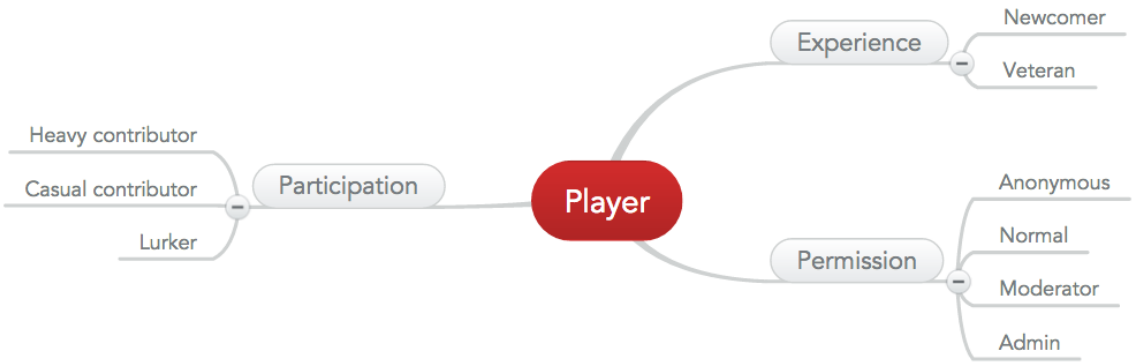


Figure 3.7: Example of player categorization.

abstraction for the state is where the player does the action to retrieve the feedback. The association to the game state could be an application screen - adapting our user story to what we call a player story. We add the state in the basic structure of the user story in the following way:

- *As a **[player]**, I want **[action]** at **[state]** so that **[feedback]**;*

Another consideration is that we must guess if there are chances that the returned feedback motivates the player (or any other player) to repeat the game interaction cycle. If there is no chance of instigating players to participate, maybe there is something wrong with the feedback or it is in the wrong state, or it is given by the wrong action, or executed by the wrong player. We call this specialization of user stories based on the Gameplay Interaction Model as player stories.

3.4 EXECUTION: IMPLEMENTATION AND TESTING

This phase receives as input the designed gameplay to implement and test. The implementation and testing are intimately related and sometimes it is wise to decide how the tests will be done even before thinking about implementation. In this section, we cover how we devised a way to implement and test. Figure 3.8 shows the steps of the execution: choose a test; define tester instructions; implement a test; and analyze the results.

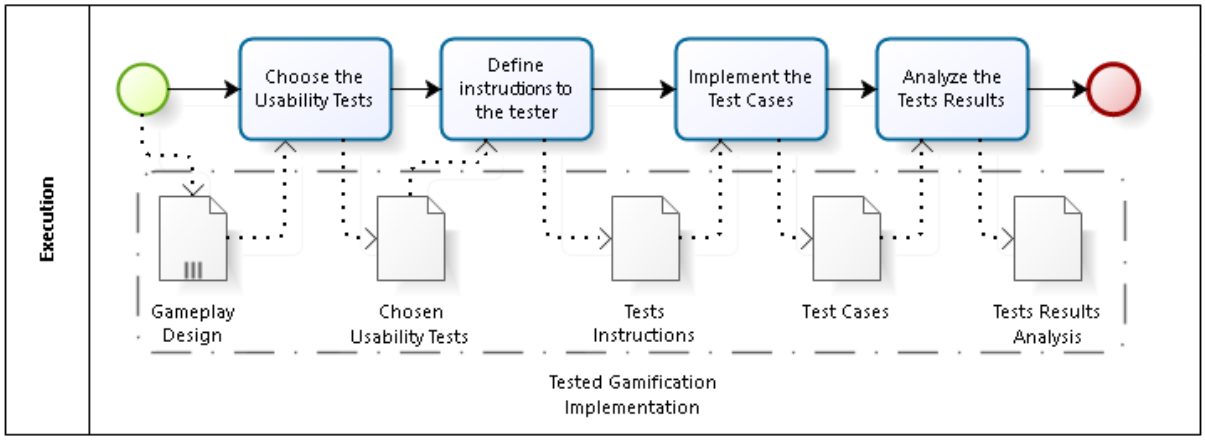


Figure 3.8: Execution sub-process.

As discussed in Chapter 2, it is better to design the user interface first because programming is more expensive and harder to change than user interface. Moreover, the user interface is what people see, interact and experience, therefore by starting with user interface we can quickly collect results from our experimental approach.

In our experimental approach we prototype functional and non-functional screens and evaluate through usability tests. This way it is possible to evaluate the application with up to 5 users and still get the best results. Another good point of this is that we focus on the user experience issue instead of being troubled by network communication problems caused by communication between the client application and the server application, for example. If the application already exists before we plan to gamify it, we can depart from the original screens and modify them according to the designed gameplay. It must simulate the use with real users in a convincing way.

3.4.1 Choose usability tests

In this step, the developer determines which usability tests, as presented in Section 2.2.4, can be used to test each player story, such as: Five Sec Test; Click Test; Navigation Test; and Preference Test. Thus, the developer thinks about the testing before implementing the solution by identifying which metrics should be measured to validate the solution.

3.4.2 Define instructions to the tester

After determining the usability test to be performed, the developer should prepare the instructions to the testers. These instructions should include a brief presentation of the application and its context to quickly introduce the tester to the application scenario. Then, the instructions should contain the task that the tester should perform.

3.4.3 Implement the test cases

This step is the implementation of the prototype itself. The prototype must contain all elements described in the instructions and should be ready to be tested by the specified usability tests. Then, with the results of the implementation, the developer is able to create and execute the specified test, providing the tester with the instructions and the prototype.

3.4.4 Analyze the test results

At this point, the developer analyzes the results of the usability tests. If results are statistically significant, the implementation should be replicated into the real application, otherwise the developer must consider rewinding to the Analysis phase.

3.5 CONCLUDING REMARKS

Gamification is the use of game design elements in non-game contexts (Deterding et al., 2011). It is important to keep in mind that using game design elements in crowdsourcing systems will not necessarily transform these applications into games. It is a strategy to improve the user experience of collaborative systems with the use of game design thinking. In this chapter we presented our proposal of a conceptual framework for systematic designing gamification in crowdsourcing systems. It is organized in four main phases: gathering; analysis; model; and execution. It supports gamification through a process that combines knowledge from software engineering, collaborative software, game design and interaction design. Using interaction design principles to model the interface first is quicker and cheaper than programming, thus the designer can understand faster if the proposed gamification approach were well accepted by the end-users.

Chapter 4

EXPERIMENTAL STUDIES ON G.A.M.E.

To verify the feasibility of our proposal, we instantiated G.A.M.E. in a crowdsourcing system in the domain of public transportation: Wikibus (Moitinho, 2013). Then we evaluated the framework usage by software developers. Section 4.1 covers the instantiation of G.A.M.E. on Wikibus and Section 4.2 covers the qualitative evaluation with software developers.

4.1 PLAYING THE G.A.M.E. IN WIKIBUS

To evaluate the defined G.A.M.E. concepts, we instantiated it in the Wikibus application - a mobile web crowdsourcing application that acts as a collaborative repository to provide public transportation data (Moitinho, 2013). Users collaboratively access and supply information about bus stops, bus lines routes, bus vehicles and bus companies. Users are also able to register occurrences regarding transportation (e.g., a bus is late, a bus stop is closed, there is a traffic jam). This section presents how we applied G.A.M.E. to Wikibus, following G.A.M.E. main steps.

4.1.1 Phase 1: Gathering

According to G.A.M.E., the first phase on gamifying an application is to gather information and to understand the application *status quo*. To do this in Wikibus, we collected information regarding its goal, functionalities, technology and issues as summarized in Table 4.1.

To build and maintain a collaborative public transportation information repository to give users the ability to provide and share information fostering the dynamic updating of the repository. Wikibus users can register and/or update information of a certain city such as: bus lines routes; bus companies; bus stops. Like most wiki-like applications, Wikibus is a CRUD (Create, Read, Update, Delete) system. It has 4 content types BUS STOP, BUS LINE, BUS VEHICLE and BUS COMPANY. A BUS VEHICLE belongs to a BUS COMPANY and supports a BUS LINE, which attends a set of BUS STOP.

Gather Goals	**Gather Functionalities**	**Gather Technology**	**Gather Issues**
Build and maintain a public transport information system repository.	Users can register and/or update information of a certain city such as: bus lines routes; bus companies; bus stops.	- Mobile web application; - Requires HTML5 compatible browser; - Client-server Architecture; - Client-side: Sencha Touch; - Server-side: PHP and PostGreSQL.	I1. It is hard to understand how to find and how to contribute with new content; I2. It is hard to trust information that anyone can change.

Table 4.1: Wikibus - Gathered Data.

Anonymous users can only search and read contents. Authenticated users can create, read and update contents from any content type. Only moderators are allowed to delete content. Wikibus is a mobile web application, so the user needs a compatible mobile device and an Internet connection. Wikibus works in mobile O.S. that has a HTML5 compatible browser. The system architecture is client-server. Wikibus's client-side was developed with Sencha Touch [1], an HTML5 based framework to develop web applications for mobile devices. Its server-side was written in PHP and uses a PostGreSQL database to store the data.

To identify the issues regarding Wikibus, the original, non-gamified, version of the application was submitted to two experimental studies. The first one was performed by Moitinho (2013) and consisted in an observation of a controlled scenario with two participants. Its objective was to analyze and get impressions about the application functionalities, efficiency and usability (Moitinho, 2013). With the purpose of measuring how much content the users were able to create during a period of time, two users were asked to register a set of information using the application (Moitinho, 2013). Users were able to register 300 bus stops, 45 bus lines routes and 18 bus companies during 8 hours (Moitinho, 2013). Since Wikibus is a crowdsourcing application, we believed that we should evaluate it with more users. Thus, a second evaluation was an observation in a quasi-controlled scenario with several real users. During 15 days we collected information from the application visitors through the analysis of action logging and on-line survey. The application got received 279 visitors resulting in 130,209 interactions and 27 survey answers. From the survey we realized that users ranged from 16 to 49, 74% male, 70% undergraduate students, and 85% from Bahia. The log registry and the survey revealed that users were able to use the application in different browsers and operational systems from various locations. From this study we identified two issues that impact on the application goal: (I1) it is hard to understand how to find and how to contribute with new content; and (I2) it is hard to trust information that anyone can change.

[1] http://www.sencha.com/products/touch

4.1.2 Phase 2: Analysis

After understanding the Wikibus scenario, this phase covers the analysis of the application regarding the crowdsourcing aspects, collaboration tasks and the relationships with the existing issues. Table 4.2 summarizes this phase.

Analyze Crowdsourcing	**Analyze Collaboration**		**Analyze relationships between Issues and Collaboration**
Build and maintain a public transport information system repository.	**Cooperation**	Content creation and updating about public transportation information	(I1) It is hard to understand how to find and how to contribute with new content.
	Communication	Users communicate with one another indirectly through created content.	(I2) It is hard to trust information that anyone can change.
	Coordination	The content moderator reviews content created by the users.	(I2) It is hard to trust information that anyone can change.

Table 4.2: Wikibus - Analysis Results.

The collaboration nature of Wikibus is explicit, i.e., the user must act on the system explicitly to collaborate. The target problem is the building of set of artifacts, in this case a structured knowledge database, and to achieve that, each user must help the creation and maintaining of the artifacts created. The Cooperation tasks correspond to content creation and update. The Communication between users happens through the content created. Coordination is carried out by a super user who works as a content moderator. Furthermore, the system coordinates content creation by complying rules on content creation and updating.

Section 4.1.1 presents two issues in Wikibus. The first issue (I1) is that *"it is hard to understand how to find and how to contribute with new content"* imply that if users do not learn quickly how to find content and how to contribute with new content, they will not be attracted to use the application. This issue creates an obstacle for new users and

makes it hard to evaluate other application functionalities since searching for a content is one of the first things a user should do. Since Wikibus is based on mobile technology, mobile users expect a simple and intuitive interface that fits well on small screens and requires only touching for interaction. The second issue (I2) is about trusting content created by unknown users. To minimize it, it is necessary to introduce mechanisms that foster trust between users and over the created content.

4.1.3 Phase 3: Modeling

The third G.A.M.E. phase is the modeling of the gameplay for the collaboration tasks chosen in the previous phase. In this phase we looked for desirable game design patterns and wrote player stories to implement the pattern in Wikibus. A G.A.M.E. player story defined is composed by a player, an action, a state and a feedback. Before choosing the game design patterns, we classified Wikibus player types and also identified Wikibus states.

Wikibus player types can be divided according to their participation, mastery and authentication. Table 4.3 correlates Wikibus functionalities for each player type according to three criterion. We expect that only contributors access actions that can create and update and lurkers just view content. Wikibus does not limit functionalities for view, create and update actions according to user's mastery. All authenticated users can access create, view and update content, except anonymous that can view content only.

		Functionality		
Criterion	**Player type**	**View**	**Create**	**Update**
Participation	Hard contributor	Yes	Yes	Yes
	Casual contributor	Yes	Yes	Yes
	Lurker	Yes	No	No
Mastery	Expert	Yes	Yes	Yes
	Novice	Yes	Yes	Yes
Authentication	Authenticated	Yes	Yes	Yes
	Anonymous	Yes	No	No

Table 4.3: Wikibus players types and capability of use functionalities.

We can represent each screen of Wikibus as state as shown in Figure 4.1. At session start the user access the Login Form, the Main Menu and the Sign-up Form. From the Main Menu he/she can also access the Login Form and the Signup Form. From the Main Menu he/she can go to Bus lines search, Bus Stops Map, Buses search and Bus Companies List. All these states also lead to the Main Menu. For each one of these states the user can go to a form that view/edit/create the respective content.

Having a better understanding of the Wikibus player types and states gave us awareness of who and where/when should be affected by the gamification. This knowledge

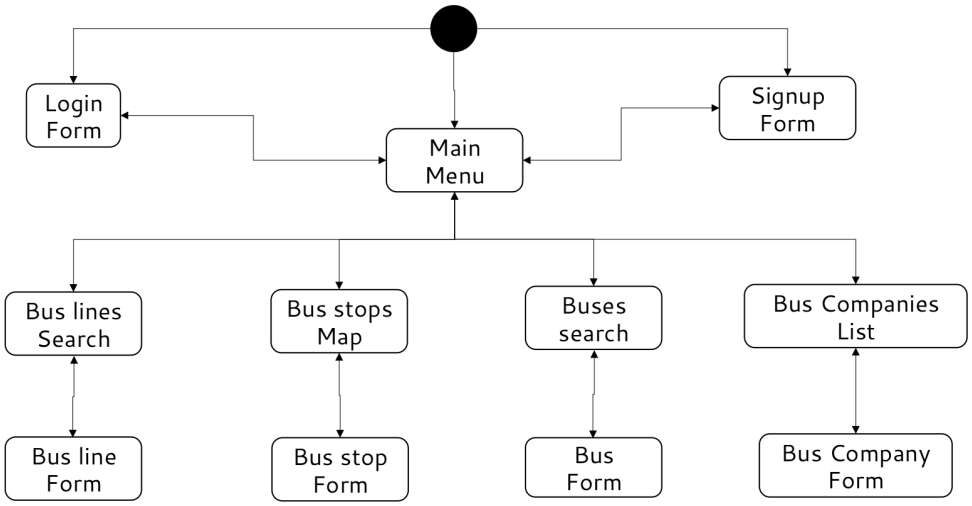


Figure 4.1: Wikibus Game States.

guided us to search Game Design Patterns that could result in desirable feedbacks in certain screens to the players of each collaboration task. We considered a desirable feedback one that could possibly solve a related issue. The Modeling phase outcomes is summarized in Table 4.4 and will be discussed in the following.

	Chose Game Design Patterns	**Define Player stories**		
Cooperation	Smooth Learning Curves	**Player**	**As**	a hard/casual novice contributor
		Action	**I want**	to learn how to create content
		State	**at**	the bus stop map
		Feedback	**so that**	I can contribute with content.
Communication	Communication Channels	**Player**	**As**	an authenticated player
		Action	**I want**	to confirm and to visualize confirmed content
		State	**at**	bus stop screen
		Feedback	**so that**	I can trust the content.
Coordination	Ownership	**Player**	**As**	a player
		Action	**I want**	to visualize who is the content creator
		State	**at**	the bus stop screen
		Feedback	**so that**	I can have a parameter to trust that content.

Table 4.4: Wikibus - Game Design Patterns and Player Stories.

We designed the gameplay for Cooperation to achieve the game design pattern Smooth Learning Curves. According to Björk and Holopainen (2005), this pattern occurs in games to provide players with the possibility of smoothly progressing from novice to expert. The Smooth Learning Curves also provides immersion and the core principle involves making players have the right level of difficulty whatever their skill levels are (Björk; Holopainen, 2005). Balancing the challenges difficulty to the players' skills can be done by providing information to players on how to overcome the challenges. The feeling that it is possible to create and contribute is the value that the application should provide for a user willing to contribute. The player is a novice as he/she does not yet know how to create content. The player is also a contributor, but at this moment we do not know if he/she is casual or a heavy contributor. The action is about learning how to create content at the bus stop map, as this is the place where he/she will have access to this feature. Originally on Wikibus, the create content feature seems to be counter-intuitive because the button to create content is not visible on the bus stop map screen. Figure 4.2a and Figure 4.2b show an example of the Bus stops map for an authenticated user. The button on the right upper corner displays a toolbar with some options for the user, including the add bus stop button.

(a) Bus stops map screen.

(b) Bus stops map screen with toolbar (after toggling the toolbar button).

Figure 4.2: Wikibus original screens of an authenticated user.

The Communication Channels pattern describes the media and methods that players can use to exchange messages (Björk; Holopainen, 2005). According to this pattern, the Wikibus Communication Channels could be: 1. mediated by Wikibus application; 2. asynchronous, players do not need to be online at the same time; 3. non-verbal, symbols or signs can be used to avoid noise and to keep the communication simple.

Our target player could be any kind of player because communication should not depend on participation, mastery or authentication. However, to send a message the user must be authenticated. The action could be related to giving the sense of trust to

a certain content, for this reason we believe that the communication could be simplified to the action of confirm (and visualize confirmation) of a certain content. To implement this feature, we could pick any Wikibus state (Figure 4.1) that shows specific content, for example the bus stop screen, as shown in Figure 4.3. This screen shows a bus stop and the related bus line routes. Our hypothesis is that a user can trust more on contents confirmed by many users than on an unconfirmed content - even if he/she does not know them. Any player can see confirmed content, but only authenticated people can confirm content. We will focus on authenticated players for now as we want to focus on confirmation. The chosen game state to implement this improvement will be the bus stop screen.

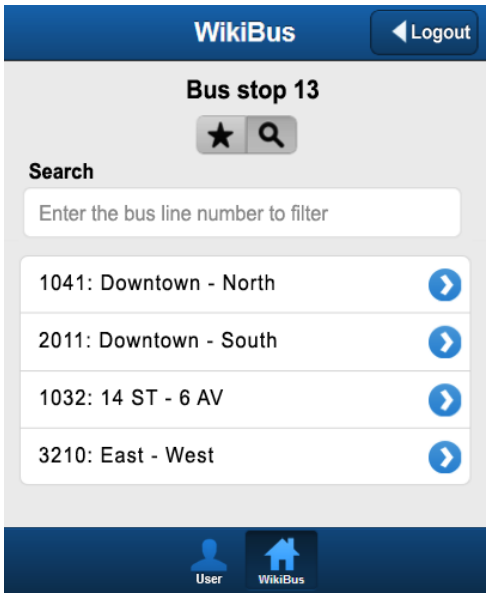


Figure 4.3: Bus stops original screen.

Finally, we searched for game design patterns to support improving coordination. In the category titled Game Design Patterns for Resource and Resource Management we picked the pattern called Ownership. This pattern is about which players have access to some components and how. Ownership can be used to motivate users to achieve goals (Björk; Holopainen, 2005). In our case, it also can instantiate the sense of Collection, so users who seek to achieve this kind of personal reward can be motivated to create content. Another benefit of owning some component would be that it instantiates Identification - parts of the game in which players identify themselves (Björk; Holopainen, 2005). By owning some content we expect that the user develop a sense of belonging and also care about the content. As a side-effect, other users could trust certain content because it was created by a highly-qualified user and not by an anonymous person. Despite being a simple game design pattern, Ownership seems to be effective at motivating users. For example, on Foursquare [2] there is the concept of mayorship, when a player has the most check-ins at a given place in the past 60 days. Mayorship, along with badges and points

[2]http://foursquare.com

motivate users to participate (Lindqvist et al., 2011). In our scenario, the player of this gameplay is any user that can visualize the name of the content creator at proper game states like the bus stop screen, presented in Figure 4.3.

4.1.4 Phase 4: Execution

The Execution phase covers 4 steps regarding the choosing a usability test, defining instructions to the test, implementing the test case and analyzing the results. Table 4.5 summarizes this phase.

	Choose the Usability tests	**Define Instructions to the tester**	**Implement the Test cases**	**Analyze the Test Results**
Cooperation	Navigation Flow Test	This is an app for collaborative transport public information. Suppose you want to add a bus stop that is missing on the map. What would you do?	Display a hint at the bottom of the screen teaching how to create content at bus stops map	30% of participants on modified screens against 20% on the original screen reached the final goal.
Communication	Preference Test	Below you see two versions of an crowdsourcing app for public transport system information. Which version would you trust more?	Show an indicator about how many people confirmed that information (asynchronous mediated non-verbal communication)	Variation 1: 8 out of 10 trusted more on the modified screen. Variation 2: 8 out of 10 trusted more on the screen where the confirmation indicator was higher.
Coordination	Preference Test	Below you see two versions of an crowdsourcing app for public transport system information. Which version would you trust more?	Show an indicator who created the content an a score	Variation 1: 8 out of 10 trusted more on the modified screen. Variation 2: 7 out of 10 trusted more on the screen where the user is not anonymous.

Table 4.5: Wikibus - Tested Gamification Implementation.

This was our first instantiation of G.A.M.E., so we followed previously known guidelines presented in (Shull; Singer; Sjøberg, 2008) to design, execute and document the Execution phase as an controlled experiment. Appendix B covers the details of the expeirment, here we summarize the main results.

The Navigation Flow was a good option to check the learnability of the screens of the Cooperation player story. As the player stories Communication and Coordination tasks

were about trust, Preference tests would be easier to check which option the tester would trust more. The instructions for the testers were written with two sentences: the first one to explain the test context, and the second to question the tester.

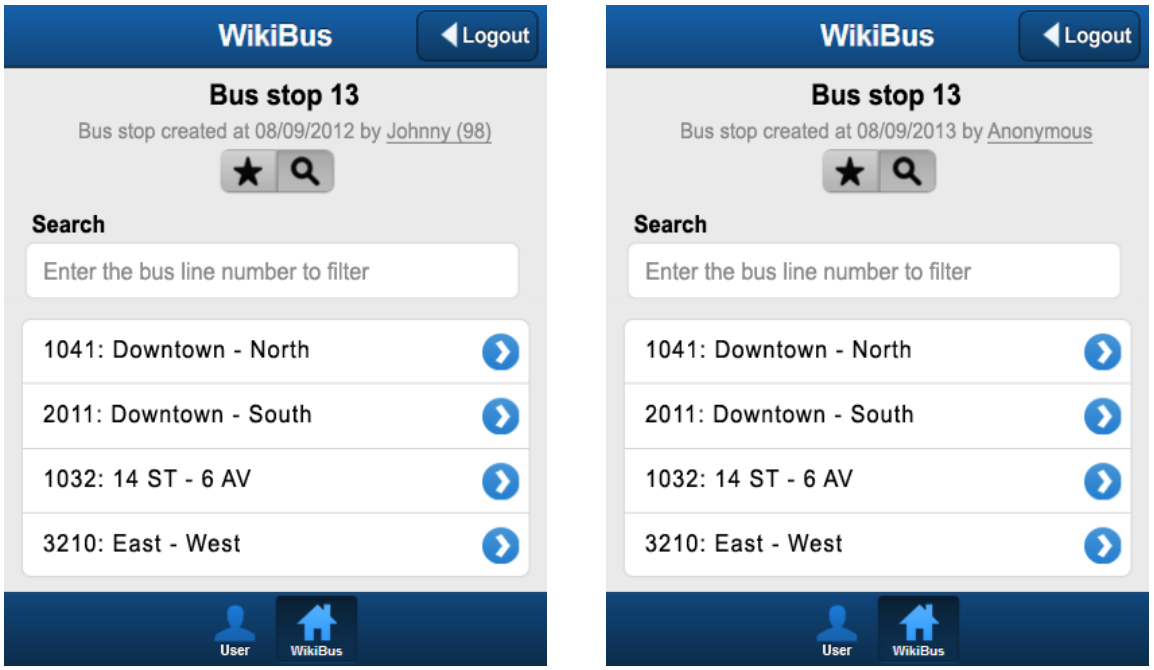
To implement the gameplay for Wikibus Cooperation task, we applied the Helpers pattern to achieve Smooth Learning Curves pattern. See Figure 4.4a. We also assume that the player is novice, authenticated and willing to contribute. This feature can overcome the problem of the counter-intuitive create content button. Communication was designed to be mediated, asynchronous and non-verbal, so we implemented the confirmation display at the bottom of the Bus Stop screen as shown in Figure 4.4b. We show a number indicating how many users confirmed that location, and by touching the number it displays a list of users and dates.

(a) Modified Bus stops map screen. (b) Bus stop screen with confirmation.

Figure 4.4: Gamified version of Wikibus Bus Stop interfaces.

Finally, coordination was designed to display the content creator as illustrated in Figure 4.5. Despite Wikibus does not allow anonymous users to create content, we wanted to compare the testers trust on content by authenticated vs anonymous users. By displaying the user who created the content, testers can differentiate the content created by an authenticated user from anonymous users. Furthermore, the number beside the username refers to its score or qualification. As a result, the tester can infer if the content can be trusted or not. A variation of this test can be executed with different scores for the content creator.

We chose to implement game design patterns that are mostly related with the nature of the collaboration aspect. We also implemented small changes to make it easier to measure benefits. Despite the result of the usability tests for the Cooperation task were not statistically significant, the results for Communication and Coordination tasks showed better performance of the gamified versions.



(a) Bus stop screen with creator with score. (b) Bus stop screen with Anonymous creator.

Figure 4.5: Gamified version of Wikibus - interfaces for anonymous user.

4.2 EVALUATING THE USE OF G.A.M.E. BY SOFTWARE DEVELOPERS

This section presents our findings on how G.A.M.E. was used by other professionals to design gamification in crowdsourcing systems. The evaluation was planned to expose software developers and designers to a situation where they could use G.A.M.E. to design gamification in a crowdsourcing system and collect their feedback. We followed guidelines presented by Shull, Singer and Sjøberg (2008) to design, execute and document experiments combined with the qualitative methods presented by Seaman (1999), Cruzes and Dyba (2011) and Saldaña (2012).

4.2.1 Goal

The overall goal of this evaluation was to examine whether the use of G.A.M.E. could help developers on designing gamification in a crowdsourcing system. Using the Goal/Question/Metric (GQM) Template we have the following statement:
Analyze the G.A.M.E. framework
for the purpose of evaluating its usage
with respect to its usefulness
from the viewpoint of software developers .

In this case, we consider the metric usefulness as the quality of achieving its goal. We wanted to measure if G.A.M.E. is useful for helping the design of gamification in crodwsourcing systems.

4.2.2 Experimental materials

The evaluation was organized in the 7 stages each one having an associated material. Table 4.6 shows materials for each stage and its goal.

Stage	Material	Goal
I	Online questionnaire	Assess participant's profile.
II	Paper reading	Provide means for the participant to understand the G.A.M.E.
III	Paper talk (20-30 min)	Reinforce the research context to answer questions/doubts.
IV	Experiment talk (20-30 min)	Present the experiment to the participant.
V	Experiment execution	Assess how the participant plays G.A.M.E.
VI	Interview recording (40-60 min)	Assess the participant's impressions of G.A.M.E.
VII	Transcription, coding and theming	Analyze the participant's impressions of G.A.M.E.

Table 4.6: Evaluation materials for each stage.

First, we sent an online questionnaire to assess the participant profile. Then, we asked the participants read a paper that presents the G.A.M.E. approach to allow them to understand it and how to use it. Then, we talked to them about the required concepts, the goal of G.A.M.E. and answer their questions/doubts after reading the paper. At the fourth stage, we instructed the participant on how to use the G.A.M.E. to design gamification for a crowdsourcing system, and then asked him/her to use it to gamify an application that they often use. Finally, we interviewed the participant to get their impressions about the usefulness of the G.A.M.E. framework. The talk and interview stages were performed either in person or with the support of online calls, according to the participant's availability. After the interviews, we performed the transcription, coding and theming to analyze the participant's impressions of G.A.M.E. The next section details how each stage was conducted.

4.2.3 Tasks

The online questionnaire (Stage 1) had 15 questions as shown in Table 4.7. The first 5 questions were to identify and understand the participant's overall profile. Then, questions Q06 to Q14 assess the participant's knowledge about the required concepts to use

G.A.M.E. The final question was asked to find out what were the participant expectations of using G.A.M.E.

ID	Question
Q01	Name
Q02	Age
Q03	E-mail
Q04	Profession
Q05	Education background
Q06	What do games mean to you?
Q07	What is your experience with game design?
Q08	What are the methods that you use to design games?
Q09	What do you know about Collaborative Software?
Q10	What do you know about Crowdsourcing?
Q11	What do you know about Interaction Design?
Q12	What do you know about User Experience?
Q13	What do you know about Usability?
Q14	What do you know about Gamification?
Q15	What would you expect from a framework for gamification design in crowdsourcing systems?

Table 4.7: Questionnaire to assess participant's profile.

After responding to the online questionnaire, we sent to each participant a six-page paper about G.A.M.E. (Brito; Vieira; Duran, 2015) which aimed to support them on understanding the purpose of G.A.M.E., its context and approach. As soon as the participant finished the reading task, we scheduled an online or in person meeting for a 20-30 minute talk to discuss the G.A.M.E. approach. The talk aimed on reinforcing their understanding of the key concepts, the research goal, and to answer their questions about the G.A.M.E.

The second 20-30 minute talk aimed to explain the experiment. We provided each participant with a gamification spreadsheet template in the form of an online spreadsheet

on Google Drive [3]. The template covers all steps of G.A.M.E. in such a way that each participant could fill in the results of each step directly on the spreadsheet. As the outcome of a step is useful as an income for another step, it would be easier to have most of the necessary information in a unique screen. For each step of G.A.M.E. we provided instructions directly on the spreadsheet in the form of notes. We also explained some necessary tools, such as the game design patterns archive (Björk; Holopainen, 2005) and usability tests[4] with UsabilityHub[5]. After this we asked the participants to use G.A.M.E. to design gamification on a crowdsourcing system of their preference. During the gamification design performed by the participant, we tracked the gamification spreadsheet to assess whether the participant was using it as expected.

As soon as they finished the gamification, we scheduled the interview composed of 8 open questions, as shown in Table 4.8. As we were performing a qualitative study, the interview had to be flexible to allow unforeseen information to be gathered (Seaman, 1999). For this reason, we asked open questions that would allow the participants to talk about their experience with minimum interference from the interviewer. Participants were asked questions with "no right" answers, as we wanted to gather opinions after the experience of using G.A.M.E. to design gamification. We asked questions to obtain information regarding the framework purposefulness, strengths, weaknesses, usefulness, threats, and possible improvements.

4.2.4 Participants

The target audience for the G.A.M.E. framework includes software and game developers as well as game designers who had some experience on software development life cycle. We invited professionals who have experience with game or software development from our academic and professional network. We looked for some diversification in the participant's profiles regarding gender, age and profession, as we expected to gather unforeseen information from them.

The participation funnel is illustrated in Figure 4.6. The column represents each participant identified by a unique number. Each row represents on stage of the evaluation. We invited 21 professionals to participate in this qualitative evaluation, 11 started the pipeline, and 4 concluded the entire process (ids 2, 3, 4 and 5).

Although we had only 4 participants that completed the whole process, we consider this a significant sample due to their diversity, as Figure 4.7 illustrates. We were able to recruit professionals from the game development and software industry and academia, from different ages, and with different educational backgrounds.

The participants had different levels of knowledge about the concepts. The participants' responses of the profile assessment were scored according to accuracy: poor, good, excellent. Figure 4.8 shows the participants' scores regarding the concepts as presented in Table 4.7. We can notice that most users had excellent level of knowledge on gamification and interaction design, and a misunderstanding on concepts for user experience.

[3]http://drive.google.com

[4]Usability tests results samples can be found in the Appendix A.

[5]http://www.usabilityhub.com

ID	Question	Expected type of information
A	Did G.A.M.E. match your expectations? Why?	Purposefulness
B	What do you like about using G.A.M.E.?	Strengths
C	What do you dislike about using G.A.M.E.?	Weaknesses
D	What do you think about applications using G.A.M.E.?	Usefulness
E	What do you see as the top risks associated with the use of G.A.M.E.?	Threats
F	How would you mitigate these risks?	Possible improvements
G	Would you use G.A.M.E. in your life? Why?	Usefulness
H	Which steps of G.A.M.E. fit perfectly with its purpose? Why?	Possible improvements
I	Which steps of G.A.M.E. don't fit with its purpose? Why?	Possible improvements
J	Would you like to mention anything else?	Additional highlights

Table 4.8: Interview questions guidelines.

This score was used to customize the 20-30 minutes paper talk to each participant, as their knowledge were different, we had to talk more about some key concepts to some participants than to others. This score is also useful for combining some analysis of the interviews to build *personas* - a model of a user who focuses on individual's goals when using an artifact (Blomkvist, 2002).

4.2.5 Experimental Design

The interview recordings were transcribed very close to the natural speaking and then coding process started. Coding means create the smallest sentence that represents the information collected. Finally, the codes found were grouped into themes to help the further analysis. Themes were created based on the content of each code, grouping then according to the codes similarities. The interviews, coding and theming phases were conducted solo by this dissertation's author.

The data collection involved the semi-structured interviews and observations suggested by Seaman (1999). We observed how the participant filled in the gamification spreadsheet and the final outcomes, including the usability test results. The observation allowed us to confirm that the participants were using the framework as expected. When we realize that a participant was stuck for more time than expected in a step, we con-

Figure 4.6: Participation funnel.

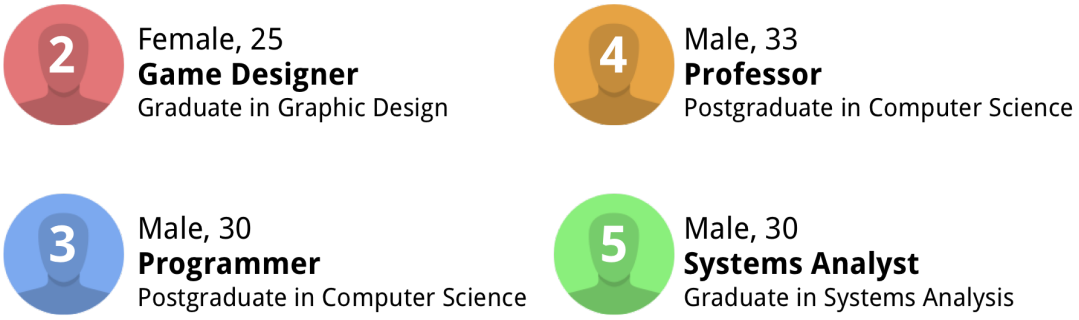


Figure 4.7: Participant's profiles.

tacted them to offer assistance. We didn't observe any serious deviation on their use of G.A.M.E., and the minor exceptions we be discussed in the Section 4.2.7.

4.2.6 Analysis procedure

We spent an average of 4 hours transcribing each 40 minutes audio recording. Then, we spent about one hour coding each transcription. The transcripts of the 4 interviews produced 20 pages of text, and an average of 26 codes per transcription. We built 15 codes patterns and organized into 4 main themes. The analysis flow of the data collected is illustrated in Figure 4.9.

After analyzing each sentence of the transcription, we extracted the data in a way that each extracted sentence had only one kind of information, feeling or opinion. Then we extracted similar sentences into codes, but preserving information to avoid interference in the data. From the original answer, we considered relevant sentences during the extraction phase. Then we created codes based on the extracted sentences. The code is the smallest sentence with the information collected from the participants. Figure 4.10 shows a coding sample.

Each code has an unique to trace its origin when needed, as shown in Figure 4.11. The

Figure 4.8: Score of the participant profiles.

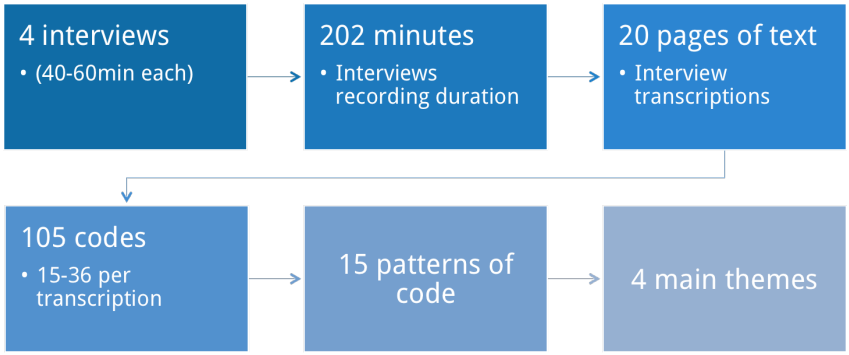


Figure 4.9: Analysis flow of the qualitative evaluation.

code ID was followed the pattern [XYZ], where: X is the participant ID; Y corresponds to the interview question; and Z corresponds to the extracted sentence. For example, the code [2A3] was the 3rd code obtained from the answer to the question A from participant number 2.

Then we identified codes patterns and organized them in main themes. Table 4.9 shows the main themes with their related code patterns and the number of codes.

The main themes are about G.A.M.E. expectations, features, phases and key concepts. To achieve the final organization of the codes, we grouped similar codes into patterns according to the code main idea. After grouping all codes according to their patterns, we named each code pattern with a category that classify all the grouped codes. Finally, we grouped the code patterns in main themes and named them using the same principle of naming the code patterns.

We classified each code as positive, mixed (or neutral) and negative. The purpose was to graphically visualize the strengths and weaknesses of the G.A.M.E. approach. Figure 4.12 shows this classification.

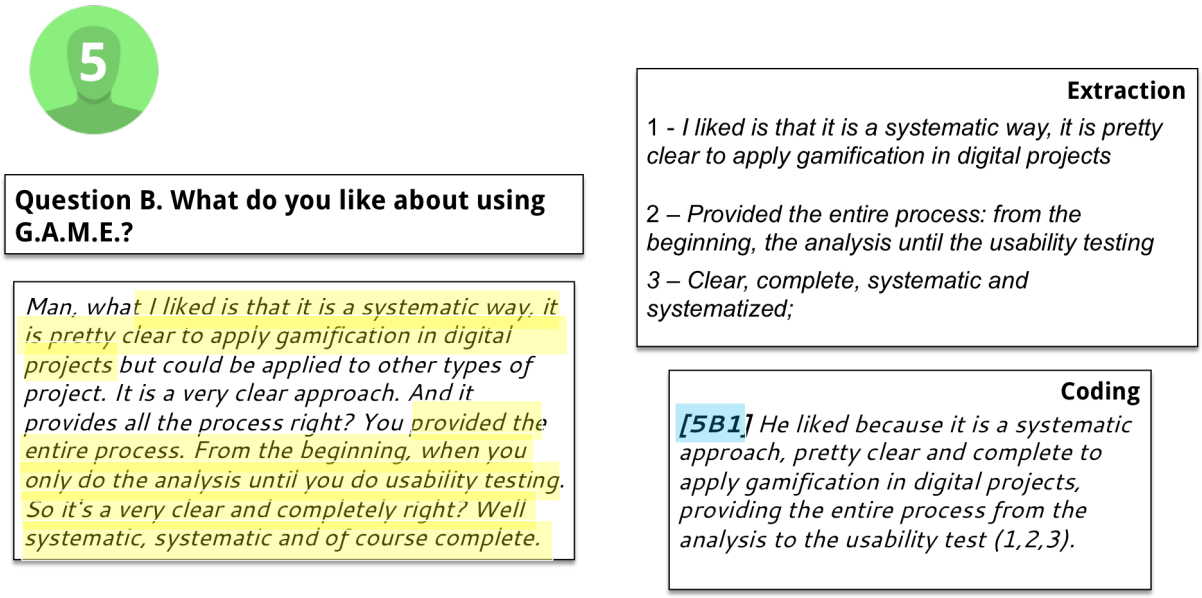


Figure 4.10: Coding sample - Original answer, extraction and final coding.

4.2.7 Discussion

This section presents a discussion of our findings based on evidence gathered from the coding process[6]. The participants said that G.A.M.E. matched their expectations. To achieve this result, we induced the participants to review their expectations before using the framework, i.e., the question Q15 of the online questionnaire. They also responded positively about using G.A.M.E. in their professional lives. For example, participant number 3 answered:

> "I believe so, because it is a visual and systematic approach with restrictions that help creativity, to raise issues and think differently [3G1]. It could be used by a group of people placing post-its, like in a business canvas, since the well-defined format helps discussions and conception of ideas [3G2]."

Their opinions about using G.A.M.E. opinions in crowdsourcing systems diverged. Some participants believed that the application could have game design elements more embedded in the system, making the system more fun and improving usability. The same participant number 3 argues that this is not enough to assure the quality of collaboration:

> "A product developed with G.A.M.E. thinks about Cooperation, Communication and Coordination, thus it has a more holistic view of collaboration and is less behavioristic [3D1]. I am not sure that the 3C and the usability tests are enough to assure the quality of the collaboration, because usability tests do not cover everything and there are important things that can only be evaluated in the real scenario [3D2]."

[6]Full transcriptions and codings can be found on the Appendices C and D

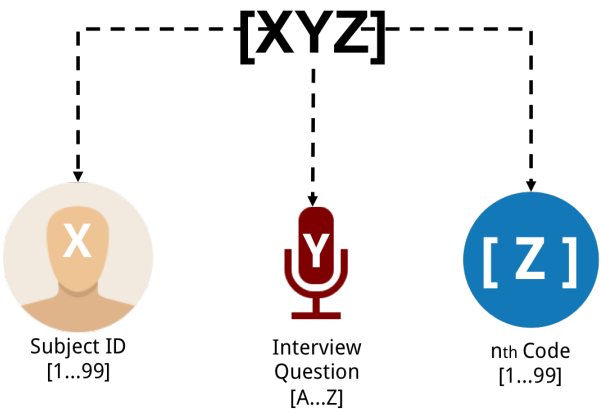


Figure 4.11: Code traceability in the coding task.

Main themes	Code patterns	Codes quantity
Expectations	Fulfillment	4
	Application	5
	Final products	6
Features	Systematic approach	11
	Cohesion	10
	Effort in use	4
	Ease of use	5
Phases	Gathering	10
	Analysis	3
	Modeling	3
	Execution	4
Key concepts	3C Collaboration Model	7
	Game Design Patterns	19
	Gameplay Interaction Model	8
	Usability Tests	6

Table 4.9: Identified themes and code patterns.

The second theme that arose from this evaluation coding was G.A.M.E. features: systematic approach; cohesion; effort in use; and ease of use. The participants had positive impressions about the systematic aspect of G.A.M.E. and its benefits. The awareness of cohesion was clear from the participants' perceptions. Most steps of G.A.M.E. were

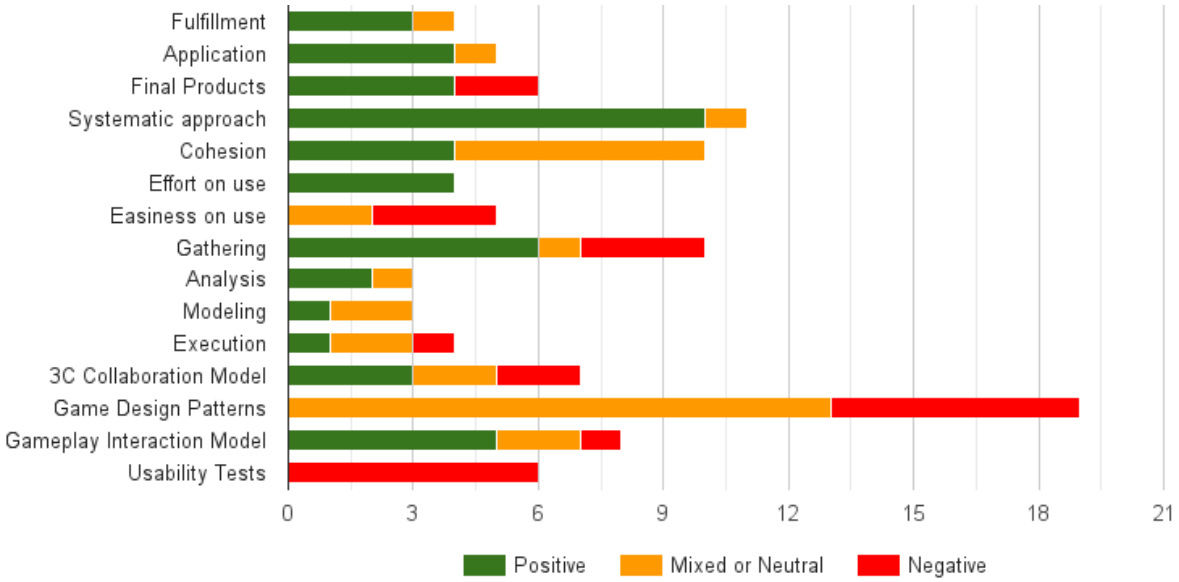


Figure 4.12: Codes classification as positive, mixed (or neutral) and negative.

useful to the participants in a way that they felt that each step was helping them to move forward. Summarizing the main positive aspects of G.A.M.E. cohesion in the participants words:

> "G.A.M.E. has well-defined steps [2A2,5A2]. Clear steps and how they relate to previous and further steps [5A3]. The whole sequence of steps made me not go directly to Execution, and to think about useful things from the Gathering phase, that helped the Analysis and that helped the Modeling [3B1]."

However, some participants had difficulty dealing with some of G.A.M.E. sub-steps. For example, the participants complained about the Gathering Technology step which at first they thought to be useful, but later they realized that, unlike the others steps, the information gathered about technology was not used in further steps. To attenuate this and other issues regarding the ease of use, the participants suggested the improvement of the documentation with examples of how to use G.A.M.E. in different scenarios:

> "Support with examples to see how other developers apply in practice is missing, it could help to situate how to get the most out of the tool [4C1, 3F2, 4F1]"

Even with some difficulty dealing with some aspects of G.A.M.E., the participants emphasized how it is generally a low effort approach to design Gamification. Combining the related reports about the effort in use, we have:

> "G.A.M.E. is a concise and objective tool with well-defined steps that are quick to execute [2G2]. The gamification process is faster than if made in an adhoc way [5G2]. It seems to be a light approach, except for the design patterns [3G3]."

The participants realized the importance of the gathered data in the further steps of the gamification design. The gathering of the individuals and community goals was highlighted by the participants as something that should be the first step, and that helped them to keep in mind this information. Participant 3 summarized the idea of the Gathering phase as follows:

> "The Gathering was important to think about the role of the application, what it expected people to do [3A4].

The Analysis phase provoked the participants to analyze the gathered data from the point of view of collaboration, more precisely to identify the crowdsourcing aspects of the application and to split the functionalities according to the 3C Collaboration Model (Fuks; Raposo; Gerosa, 2002). They understood the main purpose of this phase, however, they made two main complaints.

The first one regards categorizing the application functionalities under the 3C Collaboration Model, thinking about Cooperation, Communication, and Cooperation. They were forced to think about these 3 dimensions, and somehow felt comfortable not to think about other (unknown) collaborative aspects:

> "The G.A.M.E. is a game with well-defined rules that must be followed such as creating collaboration tasks based on 3C Collaboration Model which can give a false sense of security: Does doing well in the game means that I am doing well in real life? [3E1]. The risk is that doing well in G.A.M.E. raises issues in 3 dimensions (3C) and ignores other possible dimensions [3E2]."

This feedback suggests that further investigation is necessary to verify if the 3C Model is the appropriate to represent collaboration in crowdsourcing systems. The second issue of the Analysis phase was also related to the 3C Collaboration Model. Sometimes it is hard to categorize a task in the 3C Collaboration Model because a task could seem to fit in more than one category, for example, a task could fit in both Cooperation and Communication categories.

The Modeling phase impressions were mainly about the Gameplay Interaction Model and the use of Game Design Patterns. The use of the Gameplay Interaction Model to write user stories was very helpful to the participants. However, the use of Game Design Patterns was the step where all participants spent most of their time due to its complexity. Another unforeseen outcome was that participant 5 performed the Modeling phase unexpectedly - he changed the order of two steps, by writing the user stories before choosing the game design pattern. He argued that this way, the action and feedback from the user stories would help to find a better game design pattern. The main issue about Modeling in G.A.M.E. was the use of game design patterns. 19 out of 105 codes were complaints about how hard, complex and hard-working was to find appropriate game design patterns to design the gamification. The participants had access to an archive with 271 Game Design Patterns organized in 11 categories.

> "Game Design Patterns are complex with many variables involved, are difficult to visualize and can make the developer lose focus, using patterns that do not match the application [2E1]. The stage of Patterns was dull and it took a long time and I ended up falling into a problem that could not be implemented [3C1]."

The participants realized that the game design patterns would be more useful if they did not have so many game design patterns to study. They suggested that a reorganization of the game design patterns for gamification of collaborative systems would help them. They also thought that some game design patterns are not appropriate for collaboration, and should not be available for the developer using G.A.M.E.:

> "The list of game design patterns is very extensive and redundant, perhaps a cleaner and grouped list could help at this stage [5C4]. Patterns could be organized into smaller steps to get quick feedback of the progress in the process [3C2]. Patterns should be categorized according to the types of applications [2F1]. Patterns could have an orientation, organization taking into consideration collaborative systems scenario [3C3]. Patterns helped but gave a lot of work - low value [3I2]"

The Gameplay Interaction Model combines concepts of game theory and game design and can be used in practice as user stories from agile methodologies. The use of this model in the Gamification design was useful for the participants:

> "In the Modeling phase, the use of stories for each category was interesting to view the problem as a user [2H3]. In the Gameplay Interaction Model, the feedback helps to think about what the final purpose of the story is and how it is related with the initial goals [3A11]."

The Execution phase is the moment to finish the gamification by implementing and testing the idea. It is organized in 4 steps and was intended to guide the developer to think about the test before implementing. However, participant 5 performed the implementation stage before the stages of planning and instruction design because he imagined that the Usability test should be chosen after deciding what to implement. On the other hand, participant 2 followed the expected sequence and realized that this way he could refine the outcome of the previous step as he was progressing in the presented pipeline:

> "Execution Step divided into planning, instructions and implementation clarifies how it will work [2B3]".

An issue reported by the participants was the difficulty of using of Usability Tests. We consider further investigating if we provided poor instructions about the use of usability tests or if the tests provided so far by the UsabilityHub [7] were not enough to test their changes. Summarizing the main comments about this issue are the following:

[7] http://www.usabilityhub.com

> In Execution, some things could not be tested with interface test [3A9]. Usability tests were suitable for evaluation of changes [5A6]. The UsabilityHub does not test the necessary changes [2I3]; The risk of using the wrong usability test can result in no real results [5E3]. Instructions on how to choose the tests could reduce the developer's risk of choosing the wrong test [5F1];

From the coding we collected data about the main qualities of G.A.M.E., summarizing the data collected from the participants: a cohesive and systematic approach that helps the developer to design gamification in crowdsourcing systems. On the downside, as G.A.M.E. is a novel approach, the participants complained about some aspects. The main complaints were about how hard it is to apply some key concepts of G.A.M.E.: the 3C Model; Game Design Patterns; the Gameplay Interaction Model; and Usability tests. Appendix A presents samples of the usability tests results performed by the participants.

4.2.8 Lessons learned

Our goal with this evaluation with developers was to examine whether the use of G.A.M.E. could help developers to design gamification in crowdsourcing systems. The main conclusion is that 4 developers were able to design gamification for the first time using G.A.M.E. to improve or solve problems related to collaboration into crowdsourcing systems they knew. The qualitative-based approach allowed us to capture impressions from the participants data to confirm our expectations and revealed the strengths and weaknesses of G.A.M.E.

An important learning is related to the use of game design patterns. Table 4.10 shows the patterns used to design gamification for each collaboration aspect. A total of 21 patterns were used by the participants during the evaluation, and they applied 8, 13 and 7 patterns, respectively, to design communication, cooperation and coordination. The use of the same patterns ranged from 1 to 3 for each solution. Although we see the presence of patterns related to points, badges and leaderboards such as High Score Lists, Individual Rewards, Rewards and Score, they were not limited to those. Another expected result was that the participants used patterns very related to collaboration such as Collaborative Actions, Cooperation and Team Play.

An unexpected result was the use of patterns related to goals such as Mutual Goals, Player Defined Goals and Supporting Goal. It seems to be a result of the G.A.M.E. influence on making the participants to constantly think about the software goal on the gamification design. This result reinforces how the design of collaborative software have similarities to game design, thus, we can use game design thinking to improve collaboration.

Pattern	**Communication**	**Cooperation**	**Coordination**	**Total**
Alarms	1	1	1	3
Collaborative Actions	1	1		2
Constructive Play			1	1
Cooperation		1		1
High Score Lists		1		1
Inaccessible Areas		1		1
Individual Rewards		1		1
Leaps of Faith	1			1
Levels			1	1
Mutual Goals	1			1
Obstacles		1		1
Outstanding Features		1		1
Player Decided Results			1	1
Player Defined Goals	1			1
Resource Location		1		1
Rewards	1	1	1	3
Score		1		1
Status Indicators	1		1	2
Supporting Goal	1			1
Team Play		1	1	2
Timing		1		1
Total	8	13	7	28

Table 4.10: Patterns used during G.A.M.E. evaluation

4.3 CONCLUDING REMARKS

From our experimental studies with G.A.M.E. we could notice that our approach improved the user experience of collaboration activities with respect to their usability and sense of trust. The experimental design was conceived in such a way as to enable replication to other applications.

We performed a qualitative study to evaluate the G.A.M.E. approach with developers. The open design of the evaluation allowed us to gather information without specific hypotheses. From the results of this evaluation we have evidences of the usefulness of the G.A.M.E. approach and its most critical aspects. It also provided data to formulate specific hypotheses for future works.

Chapter 5

CONCLUSIONS

In this work, we presented G.A.M.E. (an acronym for Gathering, Analysis, Modeling and Execution), a conceptual framework to guide developers in designing gamification in crowdsourcing systems. The framework provides a step-by-step guideline that combines knowledge from software engineering, collaborative systems, game design and interaction design. This work also investigated the benefits of designing gamification by focusing on one of the main elements for user interaction: the user interface.

The main goal of this work was to investigate how game design elements could be used, in a systematic approach, to support the design of gamification in collaborative systems. To achieve this goal, we stated three research questions: RQ1) What are the main aspects to consider in the design of gamification for user interaction in crowdsourcing systems? RQ2) How can these aspects be shaped into a systematic approach for software design? and RQ3) Would such a systematic approach be useful for the design of gamification for crowdsourcing systems?

In order to answer RQ1, we investigated the main aspects to be considered to design gamification for user interaction in crowdsourcing systems. We considered knowledge from: software engineering (e.g., software development process models, user stories and prototyping), collaborative systems (e.g, individual and community goals, 3C collaboration model), game design (e.g., game design thinking and game design patterns) and interaction design (e.g., usability tests).

To answer RQ2, we built G.A.M.E. as a systematic approach to design gamification, taking as reference previous professional experiences on game design, the aspects and theories identified in RQ1, and experiments performed on building gamification into a crowdsourcing-based application for the domain of public transportation: the Ubibus-Cars (Vieira et al., 2012a).

Finally, to achieve RQ3 and verify the feasibility of our proposal, we performed two experimental studies. First, we instantiated G.A.M.E. into another crowdsourcing system - the Wikibus application. Then, to gather evidence if a systematic approach would be useful for the design of gamification for crowdsourcing systems, we performed a qualitative

study with software developers professionals using G.A.M.E. to gamify other crowdsourcing systems, such as Wikipedia, Stack Overflow, Netflix reviews and Vigilante App.

Comparing G.A.M.E. with existing works, in a previous work (Bastos et al., 2013) we performed a mapping study to investigate existing approaches to support gamification in collaborative systems. However, we could not find any method or framework, indicating a gap in systematic and integrated approaches to support designing gamification in collaborative systems. Existing commercial approaches have a reward-based design, which can be dangerous because it may result in pointing as the primary purpose of the application instead of the collaboration goal. We believe that using gamification to design interaction of collaboration activities could improve the user experience of collaborative systems.

In this sense, the G.A.M.E. approach supports developers to think about the gamification design activity, by taking the collaboration activities into consideration, and using standard methods and theories from different areas of knowledge encapsulated in a systematic guideline to support designers in a step by step design approach. From our findings, we discovered that when trying to solve problems of collaboration using game design practices, solutions were found which are common to online applications, i.e., gamification in a collaborative software do not transform the software into a game. The gamification was a result of meaningful decisions across the steps of the proposed process. We found evidence that gamification does not mean that we have to implement point systems, badges and leaderboards.

The remaining of this chapter is organized as follows: Section 5.1 overviews the contributions of this work, while Section 5.2 presents a discussion on future works.

5.1 CONTRIBUTIONS

The research presented in this work represents a step towards the definition of a terminology for gamified collaborative systems indicating concepts and activities involved in including game design elements into a crowdsourcing system. The contributions of this work are twofold: theoretical, representing advances in the literature about gamification and collaborative software; and practical, with the specification of the G.A.M.E. approach and its usage in experimental studies. In the early stages of this research, we conducted two exploratory studies with real users to map the users' needs and difficulties regarding crowdsourcing systems and to raise possible solutions to real situations Vieira et al. (2012a). Those findings can support other developers and researchers on evolving their gamification approaches. From the results found in those studies, we got evidences that rewards would motivate people to work collaboratively.

In the practical part, the principles adopted for G.A.M.E. requested integrating solutions related to the four main steps in the game design: Gathering, Analysis, Modeling and Execution. To achieve that, we proposed the integration of theories and methods from different areas of knowledge. The challenge here was to investigate how to integrate those concepts, diminishing the gap between them. This investigation produced the specification of the step-by-step activities that comprise G.A.M.E. However, to derive G.A.M.E. we had to follow a top down and bottom up approach, which led us to specify and create a crowdsourcing system to support ride sharing in a crowdsourcing

way, the Ubibus-Cars Vieira et al. (2012a) (Mascarenhas et al., 2013). As part of this path, a preliminary version of G.A.M.E. was developed, under the name of G3C - Gamification Model for Crowdsourcing, trying to solve the inequality participation problem through the use of game design elements (Brito; Vieira; Duran, 2012).

To verify if the proposed version of G.A.M.E. could be used, as proposed, we instantiated it in another case study, to gamify the Wikibus application (Brito; Vieira; Duran, 2015). This case study showed evidence of G.A.M.E. usefulness and that using game design elements in the user interfaces can improve the user experience in crowdsourcing systems. This, also, allowed us to refine our approach before evaluating it with other developers. During this instantiation we produced a spreadsheet to guide the process, giving the developer easy access to the outcomes of each step. This spreadsheet is very useful to reuse the same steps to other applications and to compare different instantiations.

To evaluate the G.A.M.E. approach with real developers, we had to create a qualitative evaluation protocol, based on Shull, Singer and Sjøberg (2008) to design, execute and document experiments combined with the qualitative methods presented by Seaman (1999), Cruzes and Dyba (2011) and Saldaña (2012). The results of this study provides a set of examples of G.A.M.E. usage, supporting other developers, and we were able to identify the most common game design patterns used to support different crowdsourcing systems. This preliminary list can support the creation of a catalogue of game design patterns for gamifying crowdsourcing systems. Tbis study showed us that even with limited knowledge in the required concepts, they were able to include new gamification thinking on their implementations. We also gathered more evidence of G.A.M.E. usefulness and its main critical aspects. In summary, participants stated that G.A.M.E. is a cohesive and systematic approach that helps the developer to design gamification in crowdsourcing systems. The analyzed data suggests that it could be improved with better documentation and refinement. For example, the use of Game Design Patterns was strongly criticized due to the number of patterns and their complexity. The Game Design Patterns should be reduced to include just the most appropriate and should be organized for gamification in collaborative systems. Better documentation would also help the developer to plan the usability tests.These studies allowed us to confirm our expectation and to realize unforeseen results that can be used to improve the G.A.M.E. experience.

Our findings on both evaluations, show the benefits of using game design thinking, through the use of patterns to solve the issues reported and to achieve the corresponding player story.

Finally, we believe that the G.A.M.E. user should be familiar with the concepts related to gamification, collaboration, and software development process. Game development experience is not required but it is a plus. Also, knowledge in programming can help to create functional prototypes for usability tests, but designers can create mockups and screen prototypes to test it as well.

5.2 FUTURE WORK

From the findings of this dissertation, we have identified several possibilities for further research.

- **Patterns for collaborative software:** One of the main complaints of the participants during the experiment with professionals was the complexity of game design patterns. Some of them also suggest that the patterns should be reorganized and categorized to be used with collaborative software. Further investigation should be performed to determine which game design patterns fit gamification in collaborative software better.

- **Testing going further than the user interface**: In this work, we conducted our gamification only over a user interface and with usability tests on computers. Although it is an easy approach to find consistent and quick results, we believe that a new study should go further. By implementing totally functional gamification, i.e., gamifying the real application instead of gamifying the user interface prototypes, we could evaluate other aspects and use other game design elements.

- **Documentation:** Better description of each phase and examples of use. As the starting point, a future work could be organize the results of the gamification performed with the participants as examples and provide them in an online repository where other developers could easily access them and collaborative.

- **Collaborative Gamification**: We provided our participants with a spreadsheet with all steps of G.A.M.E., so that they could easily fill in and access the information or decisions for each step. The visual design of the spreadsheet resembles the canvas used for business model, project development and game design. Collaborative design is one of advantages of this kind of visual approach - it enables people to work together to find a solution. A future work could be redesigning the spreadsheet as a canvas, so that people can hang it on a wall and with the help of post-its, fill it with information and decisions to collaboratively design gamification for collaborative systems.

- **Metrics for collaborative software gamification**: During the Execution phase of G.A.M.E. on Wikibus, we identified a threat regarding the construct validity. Further study should be performed to investigate other factors besides usability and sense of trust to evaluate the impacts of gamification in collaborative software. This new study would be enriched after a systematic literature review of quality factors in collaboration software.

- **Evaluating of the testing:** The Testing phase of G.A.M.E. suggests creating public tests to be executed by a crowd of testers from Usability Hub. Despite being a fast way to test results, it could be seen as an internal threat to validity. A future work could perform sets of gamification results with public tests and private tests. Thus, the results from crowdsourced participants with manual selection and

assignment of testers to treatment groups can be compared. Usability Hub could still be used to perform the usability tests, however private tests for the treatment groups can be used.

- **Gamification as a process**: The conceptual framework proposed at this work has well-defined steps with inputs that are fed by the outcomes of the previous steps. A future work could be the formalization of G.A.M.E. as a process following the BPMN[1], followed by the proper documentation and evaluation.

- **Gamification Library**: Most decisions in the gamification require programming to be implemented. With a better understanding of what the best game design patterns fit with gamification for collaborative systems, the development of a gamification library to support the gamification design could be a future work.

[1]Business Process Model and Notation - http://www.bpmn.org

BIBLIOGRAPHY

Anderson, J.; Mcree, J.; Wilson, R. et al. *Effective UI: The Art of Building Great User Experience in Software*. [S.l.]: " O'Reilly Media, Inc.", 2010.

Banerjee, P. About face 2.0: The essentials of interaction design: Alan cooper and robert reimann published by john wiley & sons, 2003, 576 pp, isbn 0764526413. *Information Visualization*, SAGE Publications, v. 3, n. 3, p. 223–225, 2004.

Bastos, A.; Brito, J.; Silva, M.; Ávila, R. Aplicação de gamificação em sistemas colaborativos - uma revisão sistemática da literatura. Available at ⟨https://goo.gl/Uv3NIM⟩. 2013.

Bell, D. The Crowdsourcing THE How To on Crowdsourcing ,. 2013.

Björk, S.; Holopainen, J. *Patterns In Game Design*. [S.l.]: Charles River Media, 2005. (Game development series). ISBN 9781584503545.

Blomkvist, S. Persona–an overview. *Retrieved November*, v. 22, p. 2004, 2002.

Brito, J.; Vieira, V.; Duran, A. G3c gamification model to crowdsourcing systems. In: *Workshop de Teses e Dissertações do Simpósio Brasileiro de Sistemas Colaborativos*. [S.l.: s.n.], 2012.

Brito, J.; Vieira, V.; Duran, A. Towards a framework for gamification design on crowdsourcing systems: The g.a.m.e. approach. In: IEEE. *Information Technology: New Generations, 2015. ITNG 2015. Third International Conference on*. [S.l.], 2015.

Burke, M.; Marlow, C.; Lento, T. Feed Me : Motivating Newcomer Contribution in Social Network Sites. p. 945–954, 2009.

Costa, A. P.; Loureiro, M. J.; Reis, L. P. Do modelo 3c de colaboração ao modelo 4c: Modelo de análise de processos de desenvolvimento de software educativo. *Revista Lusófona de Educação*, Edições Universitárias Lusófonas, n. 27, p. 181–200, 2014.

Cross, N. *Design thinking: Understanding how designers think and work*. [S.l.]: Berg, 2011.

Cruzes, D. S.; Dyba, T. Recommended steps for thematic synthesis in software engineering. In: IEEE. *Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on*. [S.l.], 2011. p. 275–284.

Cullen, R.; Morse, S. Who's Contributing : Do Personality Traits Influence the Level and Type of Participation in Online Communities. n. Cmc, p. 1–11, 2011.

Deterding, S. Gamification: designing for motivation. *interactions*, ACM, v. 19, n. 4, p. 14–17, 2012.

Deterding, S.; Sicart, M.; Nacke, L.; O'hara, K.; Dixon, D. Gamification. using game-design elements in non-gaming contexts. In: *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2011. (CHI EA '11), p. 2425–2428. ISBN 978-1-4503-0268-5.

Dillenbourg, P.; Baker, M. J.; Blaye, A.; O'malley, C. The evolution of research on collaborative learning. *Learning in Humans and Machine: Towards an interdisciplinary learning science.*, Elsevier, Oxford, p. 189–211, 1995.

Doan, A.; Ramakrishnan, R.; Halevy, A. Y. Crowdsourcing systems on the world-wide web. *Commun. ACM*, ACM, New York, NY, USA, v. 54, n. 4, p. 86–96, abr. 2011. ISSN 0001-0782.

Dubois, D. J. Toward adopting self-organizing models for the gamification of context-aware user applications. In: *Games and Software Engineering (GAS), 2012 2nd International Workshop on*. [S.l.: s.n.], 2012. p. 9 –15.

Duggan, K.; Shoup, K. *Business Gamification For Dummies*. [S.l.]: For Dummies, 2013.

Duy, C.; Hoang, V.; Wang, A. Perspectives on Crowdsourcing Annotations for. 2010.

Ellis, C. Groupware: Overview and perspectives. In: *Verteilte Künstliche Intelligenz und kooperatives Arbeiten*. [S.l.]: Springer, 1991. p. 18–29.

Ellis, C. A.; Gibbs, S. J.; Rein, G. Groupware: some issues and experiences. *Communications of the ACM*, ACM, v. 34, n. 1, p. 39–58, 1991.

Freiburg, B.; Kamps, J.; Snoek, C. G. Crowdsourcing visual detectors for video search. *Proceedings of the 19th ACM international conference on Multimedia - MM '11*, ACM Press, New York, New York, USA, p. 913, 2011.

Fried, J.; Hansson, D. H.; Linderman, M. *Getting Real: The smarter, faster, easier way to build a successful web application. 37signals*. [S.l.]: Chicago, 2009.

Frith, J. Turning life into a game: Foursquare, gamification, and personal mobility. *Mobile Media & Communication*, SAGE Publications, v. 1, n. 2, p. 248–262, 2013.

Fuglestad, P. T.; Dwyer, P. C.; Moses, J. F.; Kim, J. S.; Mannino, C. A.; Terveen, L.; Snyder, M. What Makes Users Rate (Share , Tag , Edit ...)? Predicting Patterns of Participation in Online Communities. p. 969–978, 2012.

Fuks, H.; Raposo, A.; Gerosa, M. Engineering groupware for e-business. In: *First Seminar on Advanced Research in Electronic Business (EBR'2002)*. [S.l.: s.n.], 2002. p. 78–84.

Fuks, H.; Raposo, A.; Gerosa, M. Do modelo de colaboração 3c à engenharia de groupware. *Simpósio Brasileiro de Sistemas Multimídia e Web–Webmidia*, 2003.

Fuks, H.; Raposo, A.; Gerosa, M.; Lucena, C. Applying the 3c model to groupware development. *International Journal of Cooperative Information Systems*, Singapore: World Scientific, c1995-, v. 14, n. 2-3, p. 299, 2005.

Furtado, V.; Ayres, L.; Oliveira, M. de; Vasconcelos, E.; Caminha, C.; D'orleans, J.; Belchior, M. Collective intelligence in law enforcement – The WikiCrimes system. *Information Sciences*, Elsevier Inc., v. 180, n. 1, p. 4–17, jan. 2010. ISSN 00200255.

Gartner, I. *Gartner Says by 2014, 80 Percent of Current Gamified Applications Will Fail to Meet Business Objectives Primarily Due to Poor Design*. November 2012. Available from Internet: ⟨http://www.gartner.com/newsroom/id/2251015⟩.

Hill, W. C.; Hollan, J. D.; Wroblewski, D.; Mccandless, T. Edit wear and read wear. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM, 1992. (CHI '92), p. 3–9. ISBN 0-89791-513-5.

Howe, J. The rise of crowdsourcing. *Wired magazine*, Dorsey Press, v. 14, n. 6, p. 1–4, 2006.

Huotari, K.; Hamari, J. Defining gamification: a service marketing perspective. In: ACM. *Proceeding of the 16th International Academic MindTrek Conference*. [S.l.], 2012. p. 17–22.

Janzik, L.; Herstatt, C. Innovation communities: Motivation and incentives for community members to contribute. *2008 4th IEEE International Conference on Management of Innovation and Technology*, Ieee, p. 350–355, set. 2008.

Kaufmann, N.; Veit, D. More than fun and money . Worker Motivation in Crowdsourcing - A Study on Mechanical Turk. n. 2009, p. 1–11, 2011.

Knautz, K.; Guschauski, D.; Miskovic, D.; Siebenlist, T.; Terliesner, J.; Stock, W. G. Incentives for emotional multimedia tagging. In: *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work Companion*. New York, NY, USA: ACM, 2012. (CSCW '12), p. 53–54. ISBN 978-1-4503-1051-2.

Kraut, R. E.; Resnick, P. Encouraging contribution to online communities. *Building Successful Online Communities: Evidence-Based Social Design*, MIT Press Cambridge, MA, p. 21–76, 2011.

Krug, S. *Don't make me think!: a common sense approach to Web usability*. [S.l.]: Pearson Education India, 2000.

Leyton-brown, K.; Shoham, Y. Essentials of game theory: A concise multidisciplinary introduction. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, Morgan & Claypool Publishers, v. 2, n. 1, p. 1–88, 2008.

Lindqvist, J.; Cranshaw, J.; Wiese, J.; Hong, J.; Zimmerman, J. I'm the mayor of my house: examining why people use foursquare-a social-driven location sharing application. In: ACM. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. [S.l.], 2011. p. 2409–2418.

Liu, Y.; Lehdonvirta, V.; Alexandrova, T.; Liu, M.; Nakajima, T. Engaging Social Medias : Case Mobile Crowdsourcing. p. 1–4, 2011.

Lu, X. Encouraging Participation in Virtual Communities through Usability and Sociability Development : An Empirical Investigation. v. 42, n. 3, p. 96–114, 2011.

Lund, K.; Coulton, P.; Wilson, A. Participation inequality in mobile location games. *Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology - ACE '11*, ACM Press, New York, New York, USA, p. 1, 2011.

Mascarenhas, M.; Martins, M.; Bulcao, L.; Brito, J.; Vieira, V.; Duran, A. Um estudo de caso com analise comparativa entre plataformas para aplicacoes moveis aberta e proprietaria: Android e ios. In: *XIII ERBASE - Escola Regional Bahia, Alagoas Sergipe*. Aracaju, SE: [s.n.], 2013.

Mason, A. D.; Michalakidis, G.; Krause, P. J. Tiger Nation : Empowering Citizen Scientists. p. 1–5, 2013.

Mcgonigal, J. *Reality Is Broken: Why Games Make Us Better and How They Can Change the World*. [S.l.]: Penguin Group , The, 2011. ISBN 1594202850, 9781594202858.

Michael, D. R.; Chen, S. L. *Serious games: Games that educate, train, and inform*. [S.l.]: Muska & Lipman/Premier-Trade, 2005.

Moitinho, T. d. A. *WikiBUs: Georreferenciamento Colaborativo do Sistema de Transporte Público Coletivo*. 2013.

Nicholson, S. A user-centered theoretical framework for meaningful gamification. *Games Learning Society2012: Madison*, 2012.

Nielsen, J. 10 usability heuristics for user interface design. *Fremont: Nielsen Norman Group.[Consult. 20 maio 2014]. Disponível na Internet*, 1995.

Nielsen, J. *Why you only need to test with 5 users*. [S.l.]: Alertbox, 2000.

Nielsen, J. Participation inequality: lurkers vs. contributors in internet communities. *Jakob Nielsen\'s Alertbox*, 2006.

Nielsen, J. Usability 101: Introduction to usability, 2003. *Último acesso em*, v. 17, 2010.

Norman, D. A. *Emotional design: Why we love (or hate) everyday things*. [S.l.]: Basic books, 2004.

Nov, O. What motivates wikipedians? *Communications of the ACM*, ACM, v. 50, n. 11, p. 60–64, 2007.

Pan, Y.; Blevis, E. A survey of crowdsourcing as a means of collaboration and the implications of crowdsourcing for interaction design. *2011 International Conference on Collaboration Technologies and Systems (CTS)*, Ieee, p. 397–403, maio 2011.

Parvanta, C.; Roth, Y.; Keller, H. Crowdsourcing 101: a few basics to make you the leader of the pack. *Health promotion practice*, v. 14, n. 2, p. 163–7, mar. 2013. ISSN 1524-8399.

Peng, X.; Babar, M. A.; Ebert, C. Collaborative software development platforms for crowdsourcing. *IEEE software*, v. 31, n. 2, p. 30–36, 2014.

Pimentel, M.; Gerosa, M. A.; Filippo, D.; Raposo, A.; Fuks, H.; Lucena, C. J. P. Modelo 3c de colaboração para o desenvolvimento de sistemas colaborativos. *Anais do III Simpósio Brasileiro de Sistemas Colaborativos*, p. 58–67, 2006.

Pimentel, M. et al. *Sistemas colaborativos*. [S.l.: s.n.], 2011.

Potter, A.; Mcclure, M.; Sellers, K. Mass collaboration problem solving: A new approach to wicked problems. *2010 International Symposium on Collaborative Technologies and Systems*, Ieee, p. 398–407, 2010.

Ramage, M. *The learning way: Evaluating co-operative systems*. Tese (Doutorado) — Citeseer, 1999.

Raymer, R. elearn magazine : Gamification : Using game mechanics to enhance elearning gamification. n. September 2011, p. 1–8, 2012.

Rogers, Y.; Sharp, H.; Preece, J. *Interaction design: beyond human-computer interaction*. [S.l.]: John Wiley & Sons, 2011.

Rubin, J.; Chisnell, D. *Handbook of usability testing: howto plan, design, and conduct effective tests*. [S.l.]: John Wiley & Sons, 2008.

Russ, C. Online Crowds - Extraordinary Mass Behavior on the Internet The roots : Social contagion of the "Real Crowds". 2007.

Saldaña, J. *The coding manual for qualitative researchers*. [S.l.]: Sage, 2012.

Schell, J. *The Art of Game Design: A book of lenses*. [S.l.]: Morgan Kaufmann, 2008.

Seaman, C. B. Qualitative methods in empirical studies of software engineering. *Software Engineering, IEEE Transactions on*, IEEE, v. 25, n. 4, p. 557–572, 1999.

Seltzer, E.; Mahmoudi, D. Citizen Participation, Open Innovation, and Crowdsourcing: Challenges and Opportunities for Planning. *Journal of Planning Literature*, v. 28, n. 1, p. 3–18, dez. 2012. ISSN 0885-4122.

Sessionm, I. *SessionM Drives 250 Percent Lift in Engagement Among Active Users*. October 2012. Available from Internet: ⟨http://www.marketwire.com/press-release/SessionM-Drives-250-Percent-Lift-in-Engagement-Among-Active-Users-1713446.htm⟩.

Sharp, H.; Rogers, Y.; Preece, J. Interaction design: beyond human-computer interaction. *2002*, 2007.

Shull, F.; Singer, J.; Sjøberg, D. I. *Guide to advanced empirical software engineering*. [S.l.]: Springer, 2008.

Sweetser, P.; Wyeth, P. Gameflow: a model for evaluating player enjoyment in games. *Computers in Entertainment (CIE)*, ACM, v. 3, n. 3, p. 3–3, 2005.

Usabilityhub. *Usability Hub Documentation*. https://usabilityhub.com/docs, 2015. Accessed in November 2, 2015.

Vasilescu, B.; Filkov, V.; Serebrenik, A. Stackoverflow and github: associations between software development and crowdsourced knowledge. In: IEEE. *Social Computing (SocialCom), 2013 International Conference on*. [S.l.], 2013. p. 188–195.

Vieira, V.; Caldas, L. R.; Salgado, A. C. Towards an ubiquitous and context sensitive public transportation system. In: IEEE. *Ubi-Media Computing (U-Media), 2011 4th International Conference on*. [S.l.], 2011. p. 174–179.

Vieira, V.; Fialho, A.; Martinez, V.; Brito, J.; Brito, L.; Duran, A. An exploratory study on the use of collaborative riding based on gamification as a support to public transportation. In: IEEE. *Collaborative Systems (SBSC), 2012 Brazilian Symposium on*. [S.l.], 2012. p. 84–93.

Vieira, V.; Salgado, A.; Tedesco, P.; Times, V.; Ferraz, C.; Huzita, E.; Chaves, A.; Steinmacher, I. The ubibus project: Using context and ubiquitous computing to build advanced public transportation systems to support bus passengers. *SBSI 2012*, São Paulo, Brazil, 2012.

Wake, B. Invest in good stories, and smart tasks. *http://xp123. com/xplor/xp0308*, 2003.

WAZE MOBILE LTD. *Waze - a community-based traffic information and navigation application*. 2009–2015. Available from Internet: ⟨http://www.waze.com⟩.

Werbach, K.; Hunter, D. *For the Win: How Game Thinking Can Revolutionize Your Business*. [S.l.]: Wharton Digital Press, 2012.

Xu, Y. Literature review on web application gamification and analytics. *Honolulu, HI*, p. 11–05, 2011.

Zhao, Y.; Zhu, Q. Evaluation on crowdsourcing research: Current status and future direction. *Information Systems Frontiers*, Springer, p. 1–18, 2012.

Zichermann, G.; Cunningham, C. *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. [S.l.]: O'Reilly Media, Inc., 2011.

APPENDIX A

USABILITY TESTS RESULTS SAMPLES

This appendix presents samples of the usability tests results performed for our case study and the applications chosen by the participants. These results reports from UsabilitsHub.

A.1 WIKIBUS USABILITY TEST SAMPLE

During the G.A.M.E. instantiation for Wikibus we performed a Preference Test using the original (Alternative #1) and gamified version (Alternative #2) of the Bus Stop screen. Figure A.1 shows the result report generated by Usability Hub. The instruction for the testers was: *This is an app for collaborative transport public information. Suppose you want to add a bus stop that is missing on the map. What would you do?*. The main difference between the alternatives is the confirmation information at the bottom for the gamified version. In this test, 9 out of 10 testers preferred the gamified version.

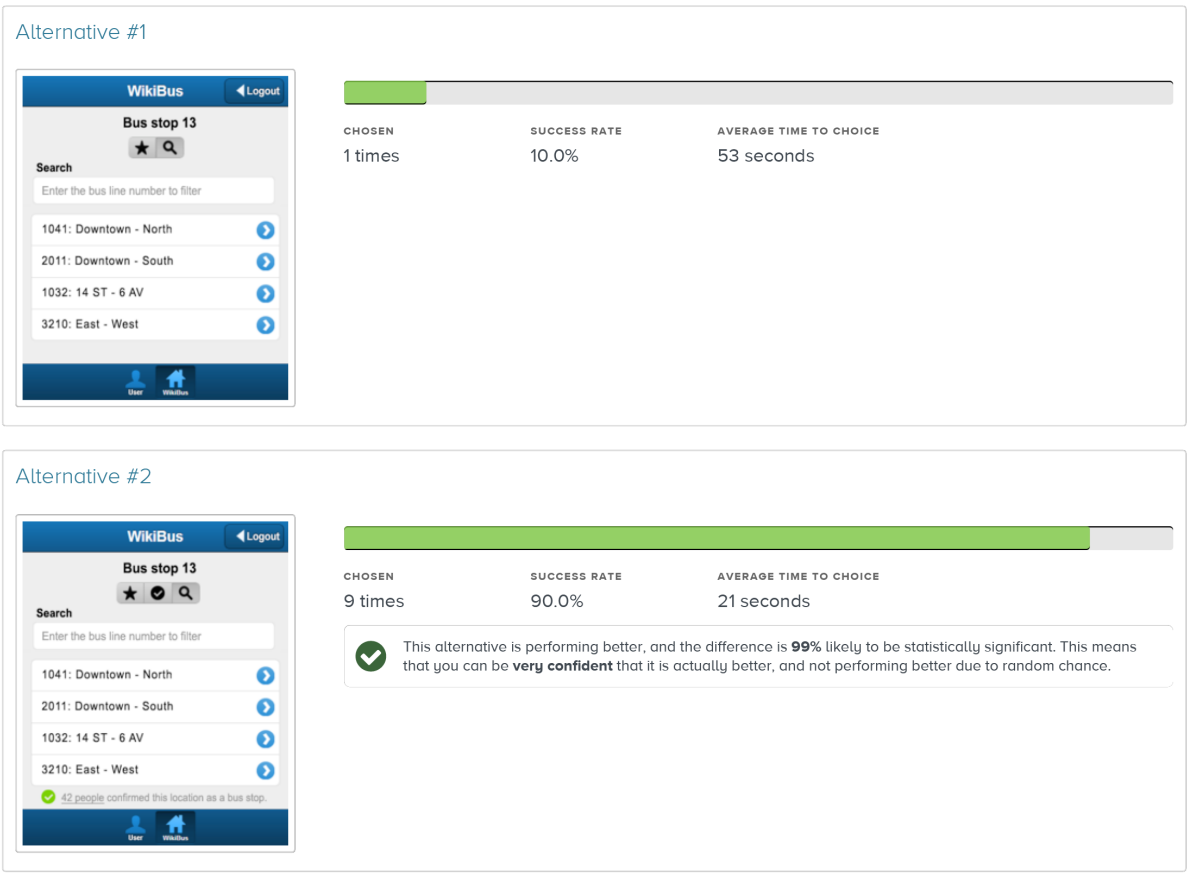


Figure A.1: Wikibus usabilityt test result sample

A.2 NETFLIX REVIEWS USABILITY TEST SAMPLE

On our qualitative evaluation, the participant 2 performed a Navigation Flow Test to evaluate the changes in the Netflix reviews. Instruction provided to tester was: *From the Home Screen, Go to the Jackass movie page, watch the movie, then rate it.* Figure A.2 shows the test variation result with the original version of the application. Only 20% of the total completed the test as expected.

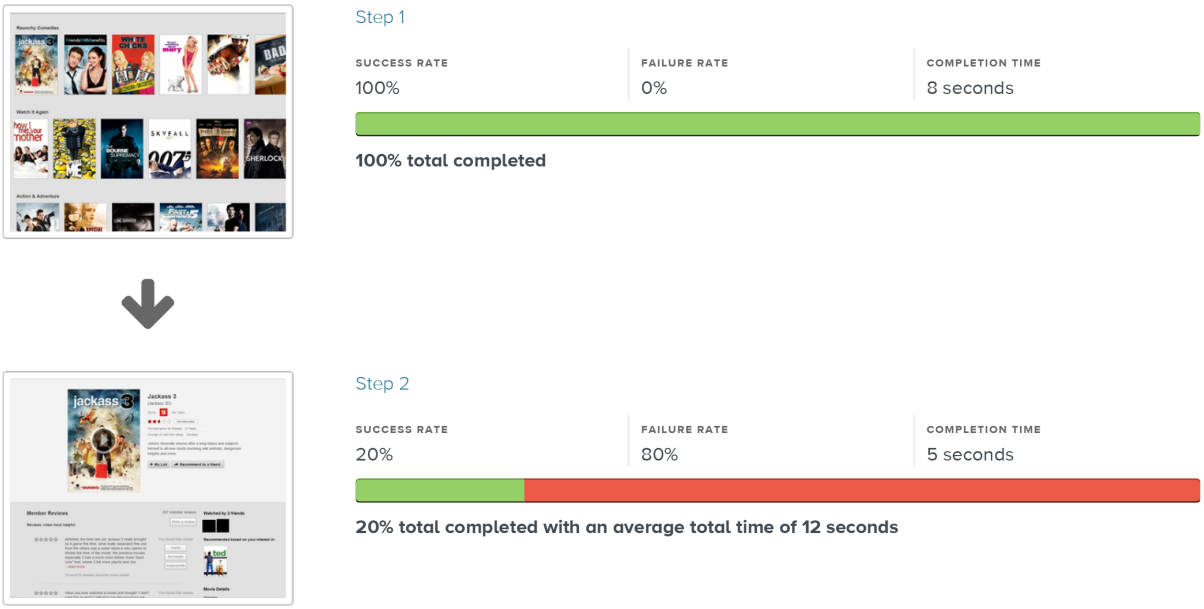


Figure A.2: Netflix reviews usability test result sample - original

The variation of the same usability test but using the gamified version is shown in the Figure A.3. In this case, 79% of the testers completed the test as expected.

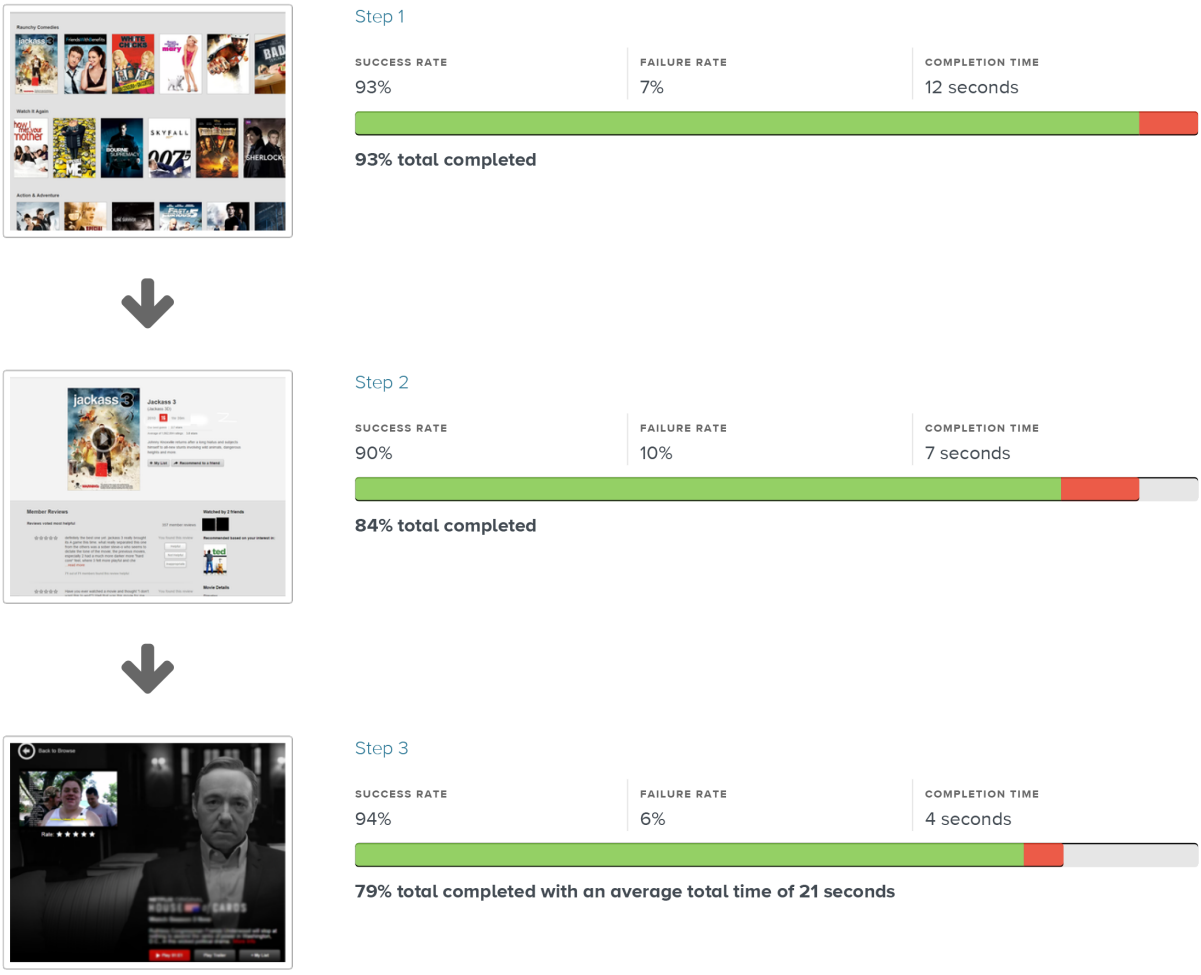


Figure A.3: Netflix reviews usability test result sample - gamified

A.3 STACK OVERFLOW USABILITY TEST SAMPLE

On our qualitative evaluation, the participant 3 performed a Click Test to evaluate the changes in the Stack Overflow. Instruction provided to tester was: *Where would you click to ask a question on this website?* Figure A.4 shows the test variation result with the original version of the application. Only 1 out of 10 testers clicked on the correct place: the "add a comment link".

The variation of the same usability test but using the gamified version is shown in the Figure A.5. In this case, 6 out of 10 testers clicked on correct places: the "add a comment" link or the "Ask for information" tab.

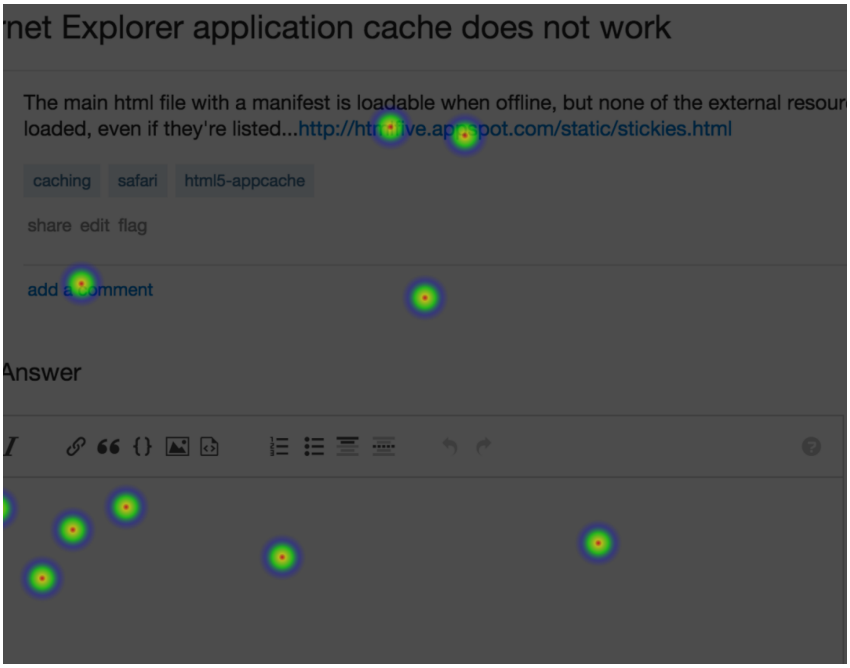


Figure A.4: Stack Overflow usability test result sample - original

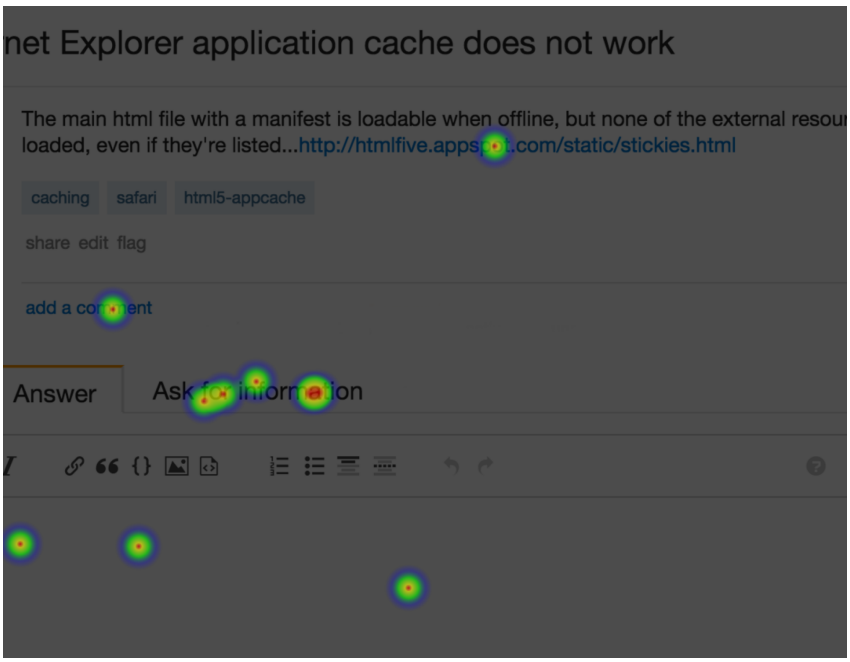


Figure A.5: Stack Overflow usability test result sample - gamified

A.4 VIGILANTE APP USABILITY TEST SAMPLE

On our qualitative evaluation, the participant 5 performed a Five Second Test to evaluate the changes in Vigilante App. Instruction provided to tester was: *This is an application that lets you report problems to the town hall in your city. You are an active participant and want to know what your position in the raking of the best contributors.* Figure A.6 shows the test result with the gamified version of the application.

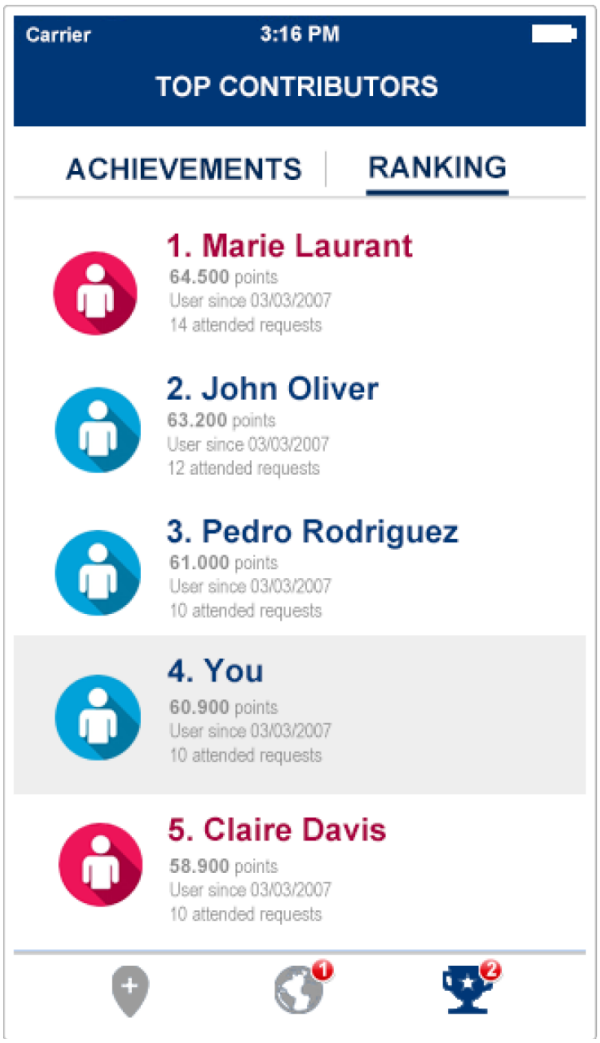


Figure A.6: Vigilante App reviews usability test screen - gamified

The result provided by Usability Hub is illustrated as a word clouds as shown in the Figure A.7. 5 out of 11 testers answered correctly the test: ”4”. The wrong answers were divided between: ”don’t know”, ”5”, top and 10.

Figure A.7: Vigilante App reviews usability test result sample

Appendix B

This appendix presents the experiment conducted to evaluate the gamification using G.A.M.E. in Wikibus

G.A.M.E. EXECUTION PHASE IN WIKIBUS

We followed previously known guidelines presented in (Shull; Singer; Sjøberg, 2008) to design, execute and document the Execution phase as an controlled experiment.

B.1 GOALS

The overall goal was to investigate how the user experience of Wikibus is affected by the gamification. Rewriting the experiment goal using the Goal/Question/Metric (GQM) Template we have the following statement:

Analyze Wikibus
for the purpose of evaluation
with respect to their user experience
from the viewpoint of testers
in the context of an online community (designers, developers and UX experts).

According to previous evaluations, the main issues are related to poor usability and lack of trust. For this reason, we defined two specific goals. As the viewpoint and the context do not change with respect to the experimental goal, we have the following goals:

Goal 1: Analyze Wikibus
for the purpose of understanding its effectiveness
with respect to the usability of collaboration activities

Goal 2: Analyze Wikibus
for the purpose of understanding its effectiveness
with respect to the sense of trust on collaboration

B.2 PARTICIPANTS

The participants of the experiment are community testers on UsabilityHub. We created public tests due to the speed of having the tests results, because usually within 1 day a test without spending time to recruit testers. We created one test for each player story, and each test was executed by 10 testers (UsabilityHub minimum). For some test we created variations (different treatments) to evaluate the original and gamified application. UsabilityHub assures that the same user just runs a test (or a variation of the same test) once.

UsabilityHub mediates the random distribution of the tests for public tests. This way, we do not control which user does each test. On the other hand, confidentiality and blinding are assured as testers do not have information about the treatments and the information provided by participants are their location, age and sex.

B.3 HYPOTHESES, PARAMETERS, AND VARIABLES

For each goal we state the null hypothesis (H_{0ij}), and the corresponding alternative hypotheses (H_{1ij}), where i is the goal identifier and j treatment identifier:

Goal 1: Analyze Wikibus
for the purpose of understanding its effectiveness
with respect to the usability of collaboration activities

Q1: Which user interface achieves a higher usability success rate?
H1: Individuals using Gamified user interface perform collaboration tasks better than individuals using original user interfaces with respect to their mean usability success rate.
H2: Individuals using Gamified user interface perform worse collaboration tasks than worse than individuals using original user interfaces with respect to their mean success rate.
H3: There is no difference between individuals performing collaboration tasks using Gamified user interface and those using original gamified user interfaces with respect to their mean success rate.

Quality Focus
Model for effectiveness for individuals: Mean usability success rate $\overline{USR}$ for all individuals, with USR = percentage of success rate for a usability test.

Goal 2: Analyze Wikibus
for the purpose of understanding its effectiveness
with respect to its sense of trust in collaboration

Q1: Which user interface is more trustworthy?
H1: Individuals using gamified user interface trust collaboration more than individuals

using original user interfaces with respect to their sense of trust rate.
H2: Individuals using gamified user interface trust collaboration less than individuals using original user interfaces with respect to their sense of trust rate.
H3: There is no difference between individuals trust in collaboration using gamified user interface and those using original user interfaces with respect to their sense of trust rate.

Quality Focus
Model for effectiveness for individuals: Mean sense of trust rate $\overline{STR}$ for all individuals, with STR = percentage of sense of trust rate after a usability test. Table B.1 overviews the variables of the experiment. $\overline{USR}$ is the mean Usability Success Rate. When a participant performs the expected behavior on task, then the task results in success, otherwise it fails. $\overline{STR}$ is the mean Sense of Trust Rate. The participant is asked to tell which version of screen is more trustworthy. Collaboration Aspect as a consequence of the $\overline{USR}$ and the $\overline{STR}$, the Collaboration aspect of each experiment is measured to verify the effect of an independent variable. The treatment performed on the user interface (Gamified or Original) is the independent variable.

Variable name	Type	Abbreviation	Scale type	Range
Treatment	independent		nominal	Gamified; Original
Mean Usability Success Rate	dependent	$\overline{USR}$	ratio	[0...100]
Mean Sense of Trust Rate	dependent	$\overline{STR}$	ratio	[0...100]

Table B.1: Experiment variables.

B.4 EXPERIMENTAL DESIGN

The design the experiment was organized in 5 stages: Test Creation; Test Variation Adding; Test Publishing; Test Stop; Results collection. On Test Creation, we picked the test type based on the kind of variable we wanted to measure. To measure STR, we picked the Click Test and to measure we picked the Nav Flow Test. On Nav Flow Tests, we first created the test with the original screen, then we added a test variation for the modified screen. On Click Tests, we showed two versions of the same screen to the user and asked the user to pick one based on the trustworthy content. We also created test variations for two modified screens, where the goal was to measure if trust would be affected if we change the trust elements.

The data collected for Click tests indicate: response click position and average response duration. The data collected for Navigation Tests indicate: success rate for each screen and average response duration. For each screen it is possible to see the response click position and average response duration just like the Click Test. Each test also pro-

vides the Tester Demographics, displaying the responses per Country, per Age and per Gender.

We performed the analysis for usability of collaboration activities (Goal 1) and the analysis for sense of trust on collaboration (Goal 2) separately. Goal 1 compares the effectiveness of two versions of the same screen with respect to the success rate of usability. For the analysis of usability, we took one independent variable: the treatment (gamified or not). Goal 2 compares the sense of trust in collaboration content of two versions of the same screen. For the analysis of usability, we took one independent variable: the treatment (gamified or not).

To measure the impact on Cooperation, the participants were told to act as if they were users of the application. However, for communication and coordination, we tried to ascertain from the users if receiving the message from communication and the result of coordination would make the content more reliable.

Collaboration aspect	Test	Test type	Task
Cooperation	#1 #2	Original Gamified	Add a bus stop in the map
Communication	#3 #4	Original vs Gamified Gamified vs Gamified	Choose the most trustworthy content
Coordination	#5 #6	Original vs Gamified Gamified vs Gamified	Choose the most trustworthy content

Table B.2: Wikibus tests tasks.

Wikibus test results for Cooperation indicate that the gamified screens have an advantage in usability, as shown in Figure B.1. For each collaboration aspect two variations were tested (original and gamified) by 10 individuals each, i.e., a total of 40 tests.

In order to check which version is most trustworthy we conducted preference tests. For each test we measured both original and gamified screens, i.e., a total of 20 tests were conducted to evaluate Communication and Coordination. Figure B.2 shows the results of the tests. This result can not be considered statistically significant, so in that case, we can not consider that individuals believe that gamified versions are more trustworthy than originals.

To double check the results of the gamified versions of Communication and Coordination tests, we repeated the tests, but this time with two versions of the gamified screen. Gamified A was the screen that was measured in the previous test. Gamified B was a version with a parameter that introduced a minor change. We wanted to know if the minor change in the parameter would decrease trust, and we could check if users were attentive

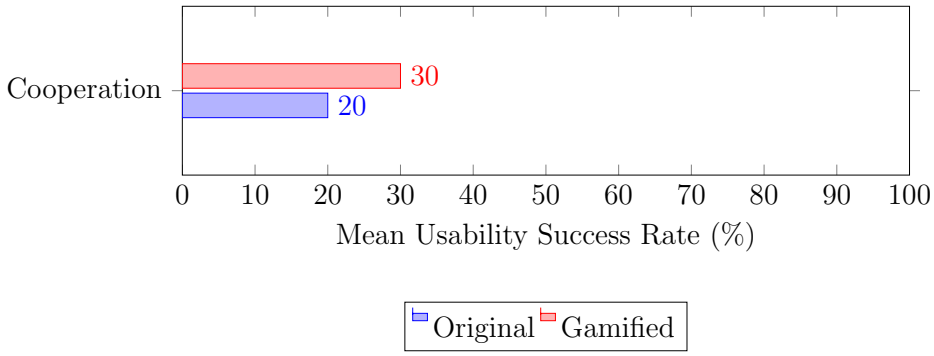


Figure B.1: Wikibus Tests results for Cooperation.

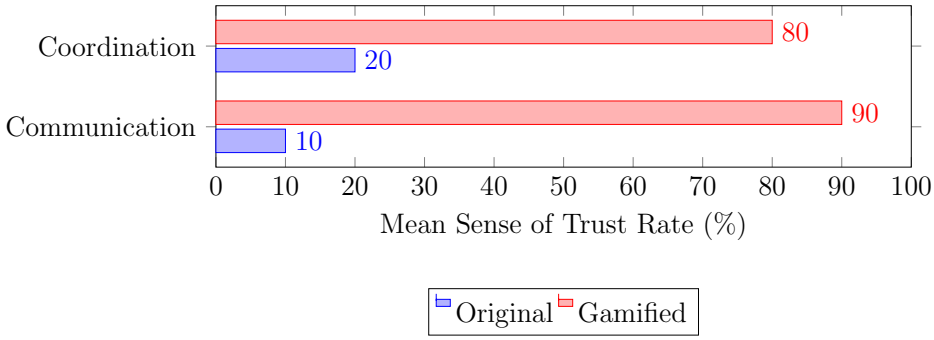


Figure B.2: Wikibus tests results for Communication and Coordination.

to the meaning of the parameter. Figure B.3 shows the test results. Again, Gamified A demonstrated that the gamification improved the sense of trust in individuals while performing collaboration tasks.

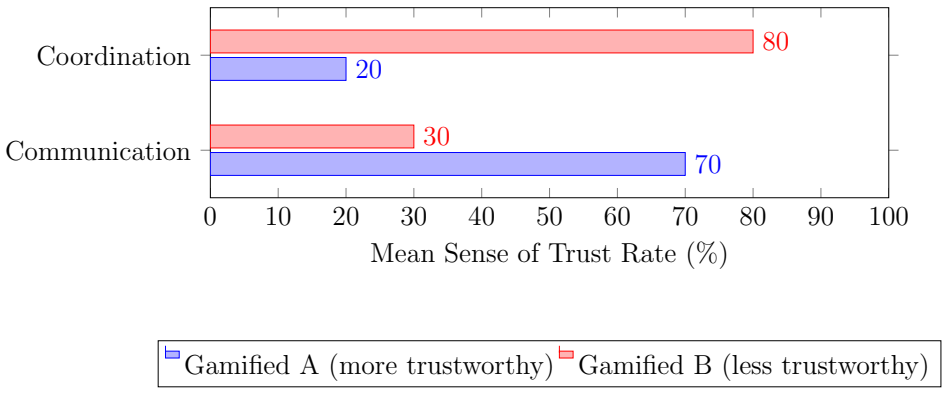


Figure B.3: Wikibus variation test results for Communication and Coordination.

Collaboration aspect	Test	Test type	Dependent Variable	Treatment	Result
Cooperation	#1	Nav Test	USR	Original	20%
	#2	Nav Test	USR	Gamified	30%
Communication	#3	Preference Test	STR	Original	10%
				Gamified	90%
	#4	Preference Test	STR	Gamified A	20%
				Gamified B	80%
Coordination	#5	Preference Test	STR	Original	30%
				Gamified	70%
	#6	Preference Test	STR	Gamified A	20%
				Gamified B	80%

Table B.3: Wikibus tests result.

B.5 THREATS TO VALIDITY

Our experience with testing user interfaces with Navigation Flow Tests and Click tests in an online community of testers resulted in consistent results. There are some possible threats to the validity of our experiment.

Construct validity.

Construct validity refers to how measures of the variables in our experiment correctly represents the respectively states in the real world. The possible construct validity identified is regarding effectiveness.

- **Effectiveness.** As there is no established definition for effectiveness in the context of gamification for collaboration, we measure effectiveness from usability success and the sense of trust based on tester behavior and opinions. These two factors were considered because they were presented as issues in the prior evaluations of the studied applications.

Internal validity.

Internal validity identifies unknown factors that may have had some influence on the results. The possible threats have been identified:

- **Assigning.** With the use of public tests, UsabilityHub assigns participants randomly. Participants takes a test and its variants once, but we do not control if the participant takes another test in our experiment. This limitation could mean that some tester participates in all of our tests, and another tester just does one test in our experiment. This way, some testers would be more experienced with the user interface than others, and somehow it could affect the results. We do not consider this a critical threat, but this could be avoided with private tests and assuring that all testers have the same knowledge of the applications studied.

- **Selection.** Participants are randomly chosen from the community, so we have no control of their knowledge of mobile user interfaces. We assume that the participants are familiar with mobile user interfaces as it is so popular nowadays. Again, this threat could be overcome with private tests with previous evaluation of their experience with mobile user interfaces.

- **Commitment.** We can not be sure that all participants were highly committed to performing our tests with attention and respect to our study. The karma-based economy of Usability-Hub could lead some testers to do random clicks just to gain karma points. Their commitment however, seems to be reasonable due to the coherence of the test results. We could avoid this threat by doing private tests with highly committed testers.

External validity.

External validity deals with issues that could prevent the findings of the experiment from being generalized to other contexts. We identified the following threats:

- **Representative participant.** As the experiment was conducted with community testers from the UsabilityHub community, we do not have control of whether the testers are potential users for the application regarding the collaboration purpose of the tested application. This could be a limitation of the external validity of our results. The ideal tester would one who would be a potential user of the application. To overcome this threat, the experiment could be replicated with private tests on representative participants.

- **Limited user experience.** Testing using screens of the user interface is an easy way to evaluate user interaction, but it is incomplete. We ignore some elements that may also affect user interaction, especially because we are dealing with mobile applications. We are ignoring mobile aspects such as the touch screen, the size of the screen of a mobile device and the context in which the user generally would use the application. In a replication of this experiment it would be wise to evaluate the application on a mobile device.

B.6 LESSONS LEARNED

We saw that gamification can increase collaboration, as usability and sense of trust are important elements in the crowdsourcing user experience. From our experiment, we could realize how small differences in the user interfaces impact on the outcome of usability and sense of trust. We also found that users thought that the content was more trustworthy if it was done by a user with high-valued number on his/her user name. Even without explaining what that number was. The participants probably inferred this number as a user interface pattern because it is used on systems such as eBay[1] and Foursquare[2].

[1]http://www.ebay.com
[2]http://www.foursquare.com

Appendix C

This appendix presents the transcriptions of the interviews of the qualitative evaluation with software developers. The transcriptions are in subject's native language: portuguese.

TRANSCRIPTIONS OF INTERVIEWS

C.1 TRANSCRIPTION OF INTERVIEW WITH SUBJECT 2

A. O G.A.M.E. correspondeu as suas expectativas e porquê? Correspondeu, porque a forma como que ele é feito, ele tem etapas bem definidas. Então acho que o que eu esperava era justamente isso. Um guia para me dizer que coisas básicas eu teria que cumprir para chegar num produto final. E acho que o G.A.M.E. faz exatamente isso. Ele te diz o que você tem que buscar no começo, o que você tem que analisar, como é que você pode pensar em resolver isso e como é que você pode testar. Eu acho que funcionou legal para o que eu esperava.

B. O que você gostou ao usar o G.A.M.E.? O que eu gostei... O que eu gostei mais do uso da ferramenta foi da etapa de Modeling, eu nunca tinha trabalhado com essas estórias, então eu achei que elas ajudam bastante para você ver o que você precisa e como é que você vai chegar nisso. Eu achei que é bem legal, é mais interessante do que você ficar lendo um monte de coisas, de problemas que você tem, é mais interessante você ler uma estória, tipo, ter um sujeito, ter um problema e o que ele esperar, é mais importante do que listar um monte de coisas que tem errado e tentar resolver isso. Eu acho que isso foi uma das partes que eu mais gostei. Nunca tinha trabalhado desse jeito. Uma coisa que eu gostei foi essa distinção na parte da Execução que você tem do planejamento, instrução e implementação. Ela faz com que você definir essas coisas que são parecidas, mas não são iguais. Te dá uma clareza melhor de como aquilo vai funcionar. Acho que tentar entender como fazer aquilo que eu tinha pensado e colocar ele no teste, me ajudou a clarear as minhas ideias, entre fazer a instrução e dizer como seria implementado. Acho que essa parte de Execução foi um, ... a parte de Modeling e Execução foram as coisas que eu mais achei bacana assim porque eu nunca tinha feito uma sequência dessa completa assim. Saber o que que eu preciso, planejar, criar essas instruções do que vai ser executado, implementar e testar. Acho que essa sequência é

muito legal. Eu acho que a sequência de tudo é interessante, mas a parte de Gathering e Analysis as vezes não é tão clara. Acho que quando você parte para o Modeling e para o Execution você começa a entender mais aquilo que você queria com Analysis e com o Gathering. Acho que as vezes a ideia não está muito clara no começo, e quando você passa para a parte da estória e do planejamento das instruções você começa a de fato clarear a sua própria ideia, as etapas como estão feitas principalmente em Modeling e Execution acho que ficou fluido, ficou bacana.

C. O que você não gostou ao usar o G.A.M.E.? Uma parte que eu não gostei muito foi relacionado um pouco a Patterns, porque? Apesar delas serem muito uteis, elas são um pouco complexas para ser usado num...., é porque o G.A.M.E. fica muito dependente de você saber isso. Então acho que talvez ele fique um pouco engessado pelas Patterns, se você não compreende elas bem você não consegue avançar. Eu não sei se teve coisas que eu gostei ou não gostei fora isso, mas teve coisas que eu tive mais dificuldade. As coisas que eu tive mais dificuldade foi dizer o que que era cada elemento, nomear essas coisas, o que é que é uma issue para mim, o que que é uma tecnologia, acho que essas coisas foram um pouco difíceis de entender no processo, demorou um pouco. Eu acho que talvez a, por não estar familiarizada com a as nomenclaturas das coisas foi um pouco difícil para mim de entender o que elas significaram e o que que isso interferia no que eu estava tentando fazer, o que que era muito importante eu saber com certeza o que era e o que que não era assim tão gritante se eu compreendesse mais ou menos. Acho que a explicação das coisas não estava muito clara as vezes. Essa coisa de dividir cooperação, comunicação e coordenação também eu acho que é um pouco confuso as vezes, porque tem coisas que se aplicam para mais de um, aí eu acho que é meio complicado de você tentar decidir como classificar isso. O que que é o que.

D. O que você pensa sobre aplicações usando o G.A.M.E.? Eu acho que eu espero que as funcionalidades que elas tentam implementar que tenham a ver com gameplay, com jogo sejam mais integradas ao sistema como um todo do que adições para parecer que aqui é um jogo sem de fato ser. Acho que não sei consigo explicar isso direito assim. Espero que não seja uma coisa que eles tinham um aplicativo pronto e pensaram assim "Poxa! Se a gente colocar gamificação aqui vai ficar legal e a gente vai vender mais" E eles tentaram implementar alguma coisa depois daquilo já ter sido tudo pensado e já ter fechado. Eu espero que por ser uma coisa que foi usado o G.A.M.E. na sua concepção ele seja o gameplay e a gamificação seja mais integrada ao sistema e não só adicionando coisas em lugares que eles acharam que funcionaria colocar ponto ou colocar score ou colocar badge, enfim.

E. Quais são os principais riscos associados ao uso do G.A.M.E.? Eu acho que pensando pelo meu projeto que eu fiz, eu acho que talvez você... a pessoa tente colocar funções ou usar patterns que não caibam com o tipo de aplicação que você está desenvolvendo. Porque assim eu acho que quando você usa as patterns elas são legais porque é uma coisa que já está lá e que você tem uma descrição de como funcionam e

que em teoria já foram testadas, então não é nada mirabolante que você precise fazer algum tipo de aprovação primeiro, mas que também pode te desvirtuar de alguma coisa que talvez não seja necessária no jogo, no jogo não, na aplicação. Por exemplo, no meu caso, eu acho que esse negócio de você ganhar ponto e votar em coisas é uma solução legal e genial para um problema que tá lá que é a falta de confiança de que você vai ter um review interessante, porém talvez seja uma coisa que tá deixando muito aberto pra você captar coisas que você não vai poder retornar pra sua comunidade. Acho que tem alguns tipos de patterns, alguns tipos de coisas que você pode criar com o G.A.M.E. que talvez você não consiga visualizar que você não possa tender. Você acha que a pattern é boa porque é um padrão, mas ela é ruim porque se você não souber usar você acaba se perdendo. Tem muita variável, muita coisa, e uma coisa vai levando a outra e a outra, e é muito complexo.

F. Como você atenuaria esses riscos? Essa é complicada! Eu acho que se existisse uma maneira. Eu não sei se isso seria possível. Mas se existisse uma maneira de você meio que diminuir o nível de complexificação das possibilidades que você tem, dependendo do tipo de aplicação que você está fazendo. Por exemplo, se você tivesse fazendo uma aplicação como Waze e você conseguisse diminuir o número de patterns que seriam acessíveis para esse tipo de aplicação, eu não sei se isso é possível, pois talvez ajude, porque tem algumas coisas que não são aplicáveis a todos os tipos de jogos ou todos os tipos de aplicações. Por exemplo, existem funções que são especificas de um adventure e existem funções que são especificas de um M.M.O, como existem funções que são especificas de um aplicativo de transito e funções que são especificas de um aplicativo de chat, por exemplo. Eu acho que se fosse possível fazer essas categorias de coisas que são mais usáveis no tipo de aplicação que você está fazendo eu acho que facilitaria um pouco talvez o entendimento e a combinação dessas coisas para você não se perder num mundo imenso.

G. Você usaria o G.A.M.E. em sua vida? Por que? Eu acho que provavelmente sim porque eu acho que o G.A.M.E. é uma ferramenta bem sucinta assim, ela é bem objetiva, você tem etapas muito bem definidas e não são etapas que levam muito tempo para serem feitas. Eu acho que as coisas que tive mais dificuldade eu levei menos de 40min para fazer. Então eu acho que se você tem uma coisa que talvez precise ser resolvido com agilidade, ela é uma ferramenta interessante de se tentar usar, porque eu acho que o fator tempo de produção dentro do G.A.M.E. é uma coisa muito interessante. Eu acho que assim que coisas que se eu não tivesse um método como o G.A.M.E. eu levaria muito mais para pensar e acho que levei muito menos tempo tendo pontos individuais e objetivos para responder, apesar de algumas coisas serem difíceis de entender, ainda assim ele me deu agilidade na construção das minhas ideias e na execução.

H. Quais etapas do G.A.M.E. se encaixam perfeitamente em seu propósito? Por que? Eu acho que é importante separar um pouco essa parte dos objetivos, objetivo individual e o comunitário. Porém as vezes eles podem ser muito iguais, as vezes um

objetivo que é comunitário ele também é um objetivo individual. Porque, por exemplo no meu exemplo querer melhores recomendações é um objetivo que todo mundo gostaria, mas é uma coisa que eu como usuário individual preciso ter para que a minha experiência no aplicativo seja interessante. Então assim, eu não estou fazendo aquilo só porque eu vou colocar mais informação na comunidade e isso vai ajudar todo mundo, mas porque eu também quero poder confiar também naquela informação que está ali. Aí eu acho que eles se confundem um pouco mas são muito importantes. O objetivo sim, acho que lugar que ele está é importante porque antes de resolver o problema eu preciso saber com o que que eu estou lidando. Então acho que a primeira coisa que você tem que saber é: o que é, que plataforma é essa, o que que você está fazendo, "ah eu estou trabalhando com streaming, eu estou trabalhando com transito, eu estou trabalhando com segurança pública". E o que que essa plataforma pretende, o que que ela está trazendo para os usuários, como objetivo individual e da comunidade. Acho que isso é muito importante estar no começo porque eu preciso entender o que é que eu estou trabalhando para tentar resolver os problemas. Eu acho que essa separação do Modeling em estórias para as categorias foi uma coisa que ficou bem colocada assim no processo porque depois de você conseguir separar o que que são seus problemas se colocar como usuário tentando resolver esse problema antes de tentar planejar e executar como desenvolvedor é uma forma interessante de ter perspectiva naquilo de que está fazendo. Porque as vezes como desenvolvedor eu acho que aquilo é um problema que tem que resolver de um jeito, mas eu não imagino como é que o usuário enxerga aquilo. Então tipo fazendo a parte de Modeling eu alterei um pouco também o que eu estava pensando em pattern, então acho que assim a colocação assim da sequência das ações ficou bem legal.

I. Quais etapas do G.A.M.E. não se encaixam perfeitamente em seu propósito? Por que? Eu acho que essa parte de tecnologia é importante para você poder pensar na aplicação das coisas porque é diferente para a plataforma que você está usando, mas talvez ela não seja tão importante para tentar entender uma função de resolver o problema que você tem. Eu acho que a tecnologia não seria tanto o Gathering mas uma pós-produção. Pelo menos para mim foi assim. Acho que função eu posso implementar aqui? Ah massa! Eu posso implementar essa função. Ok, agora qual o suporte que eu tenho para implementar essa função. Na parte de prototipagem, implementação, você pensar onde o seu problema seria mostrado e resolvido. Eu sei que é uma parte importante você saber onde é, mas foi uma coisa que eu não pensei muito até o final. A tecnologia eu acho que talvez não tenha se encaixado, não no lugar que ela está. A parte de issues, crowdsourcing e tasks, talvez elas sejam um pouco repetitivas pelo fato de você primeiro listar e depois separar. Talvez, esse issues e esse related issues não precisassem ser duas coisas. Talvez só precisasse ter cooperação, comunicação e coordenação e as issues que você tem quanto a isso. Porque eu acredito que elas teriam que se encaixar em alguma categoria. Então, talvez tentar colocar essas classificações já essas issues por categoria, fique claro de você entender quais são os problemas e onde eles estão colocados. Estaria pulando o final de Gathering e indo direto pra analysis, talvez não, ou talvez já tentar misturar um pouco o Gathering e o Analysis e já fazer esse final do Gathering já separando um pouco as coisas. Eu acho que a última coisa, só a parte de Implementing e Results eles, eu não

sei se usar a ferramenta de usabilidade que foi usada para esse teste atendeu ao nível de interferência que eu tentei fazer no sistema. Então talvez não tenha sido uma dificuldade do sistema em si, mas de como ele foi testado. Eu tive um pouco de dificuldade nessa parte de conseguir adequar o que eu precisava passar para a ferramenta para o teste. Então talvez tenha sido um problema da ferramenta que foi utilizada apesar de eu achar que o UsabilityHub é bastante bacana assim, é simples é fácil de fazer e fácil de testar. Mas talvez ele não atenda a todas as necessidades do que eu projetei ou eu não soube colocar direito.

C.2 TRANSCRIPTION OF INTERVIEW WITH SUBJECT 3

A. O G.A.M.E. correspondeu as suas expectativas e porquê? É, acho que sim, acho que foi de acordo com o que eu estava esperando mesmo. O que eu acho que eu mais foquei foi que me ajude com boas práticas né? Que me oriente para fazer as coisas no caminho certo e acho que essa orientação teve sim, principalmente quando você divide em Cooperação, Comunicação e Coordenação né? Me faz pensar em coisas que talvez eu não pensaria e que são importantes. Ah, minha expectativa era que me ajudasse com boas práticas a fazer coisas que fossem boas práticas e eu acho que o GAME ajudou principalmente quando dividiu em Cooperação, Comunicação e Coordenação que fez obrigatoriamente a pensar em aspectos que talvez eu não pensasse se fizesse sem isso. Acho que o Catering foi importante para refletir melhor sobre o papel do site o que que a gente está tentando obter com o site o que que as pessoas fazem no site, então ajudou a ter isso claramente. Assim essa parte do padrão foi a parte que eu mais demorei e assim foi uma experiência diferente, inusitada, né? Demorou um pouco para eu me acostumar porque os padrões são todos falados em termo de jogador e de jogo né? E aqui eu penso na colaboração e as vezes eu acho que mais ficou diferente é que no jogo você tem o espaço do jogo, o mundo do jogo e aqui você tem uma interface gráfica, uma GUI, então é um pouco diferente porque quando você fala em lugar em jogo, no mundo físico é um meio diferente de um lugar na interface. Ah, ?um lugar privilegiado de onde você pode ver os outros jogadores?, como é que é isso na interface? Algumas coisas não ficaram um mapeamento tão claro. Mas, ao mesmo tempo ver sob essa perspectiva de jogo faz você pensar em coisas diferentes também né, faz você ver a colaboração e o site sob uma outra ótica. Que eu acho que é bom para a criatividade também. Execução, o que eu percebi foi que algumas coisas que eu? na verdade o terceiro ponto aqui da Modelagem né que foi de coordenação, eu acho que não conseguiria usar usando teste de interface né? Porque eu acho que tem algumas coisas você tem que ver o site rodar e ver o resultado, quantas questões foram fechadas... por exemplo isso né comparar duas métricas: quantas questões foram fechadas antes e depois de fazer a mudança no site. Não consegui executar o teste para coordenação. O gameplay interaction model ajudou, assim, o player e o action são bem diretos do issue, vem direto do issue, e assim o state ajuda a pensar por exemplo em que página você trataria isso, então tem algumas coisas que eu coloquei na página inicial, aí a outra eu coloquei na página da questão, isso foi bom e o feedback também ajudou a pensar qual o meu objetivo no final das contas e o objetivo que vai, que está de acordo com os objetivos que eu listei no início, dos goals. Que aí isso me ajuda a ver

como eu vou medir, tendo bem claramente qual é o objetivo eu sei como é que eu vou medir. Tanto é que o último objetivo, o terceiro lá eu vi que era uma coisa que eu não ia conseguir medir.

B. O que você gostou ao usar o G.A.M.E.? Eu acho que é um pouco do que eu já falei, eu gostei dessa coisa sequencial que me ajudou a pensar coisas no início que iam ser uteis, por exemplo o Gathering foi útil pra Analysis, e Analysis foi útil pra Modelagem, porque eu como programador eu quero ir logo pra Execução, e o GAME me induziu a não fazer isso e acho que afetou positivamente o resultado. E essa coisa dos 3Cs acho que foi legal também.

C. O que você não gostou ao usar o G.A.M.E.? Bom, eu demorei muito tempo no Patterns e demorar muito tempo em uma coisa, eu não gosto de demorar muito tempo. Então eu fui ali numa sequência, 3 minutos, 5 minutos, 3 minutos, 6 minutos, 1 minutos, aí daqui a pouco 1 hora e 8 minutos. Se talvez, talvez se isso pudesse ser mais destrinchado, ou tivesse alguma orientação que... porque também eu estava olhando aquele guia de padrões mais ou menos pela primeira vez sozinho, então eu achei que essa parte eu fiquei muito solto e isso não foi muito legal. Assim eu vejo, eu não sei se isso seria possível, eu estou pensando no problema, o problema é que eu demorei muito tempo nisso aí e isso demorar muito tempo é ruim, principalmente quando você vem com uma expectativa de fazer as coisas em poucos minutos e pula para uma hora. Então poderia ou quebrar isso em várias etapas menores para eu ter um feedback rápido de que eu estou avançando no processo, porque aqui demorou muito para eu ver se eu estava avançando ou não. Não sei se é possível quebrar isso em mais etapas, mas seria uma estratégia. Outra é, se tivesse alguma orientação, uma organização dos padrões pensada nesse cenário de sistema colaborativo, talvez ficaria mais fácil também. Foi maçante essa tarefa de achar os padrões, talvez também porque, assim toda essa parte de Analise e Modelagem em conjunto eu estava tentando achar problemas que talvez eu conseguisse, porque eu já sabia como era a execução, então eu pensava no problema pensando também como eu iria executar né, pra não cair num problema que eu não conseguiria executar, e acabei caindo nesse problema de qualquer jeito, tanto é que se você for ver os issues que eu levantei na Seção 1 são diferentes dos que eu levantei lá na Seção 2 de Analise, justamente porque esses que levantei inicialmente eu percebi depois que não iria dar pra testar. Por exemplo, eu pessoalmente como usuário do site, como pessoa que responde perguntas, um grande problema que eu vejo é essa primeira issue que eu levantei no 1.4 que é, ou quando alguém faz uma pergunta assim fácil, 30 segundos depois já tem uma resposta. Então eu não sou incentivado a escrever uma outra resposta ou até escrever uma resposta melhor porque quem chega primeiro já ganha os pontos, já está em vantagem na corrida. Aí eu até em um mecanismo de sei lá o site esconder de mim que já tem uma resposta praquilo ali durante um tempo para eu ficar incentivado a escrever minha resposta. Mas aí pensando nisso como teste de interface, já é mais complicado.

D. O que você pensa sobre aplicações usando o G.A.M.E.? Bom, se eu souber que um produto foi desenvolvido usando esse GAME aí. Isso, como é que eu vou ver esse produto sabendo disso né? Assim, eu vou saber ou pelo menos achar que ele pensa em Cooperação, Comunicação e Coordenação, portanto tem uma visão um pouco mais holística da colaboração. Um pouco menos behaviorista, de toma o queijo, ou toma a cenoura né que o pessoal fala e anda para a frente, ao mesmo tempo eu não estou seguro assim que essas três coisas Cooperação, Comunicação e Coordenação, e o uso de testes de interface elas são abrangentes o suficiente para garantir a qualidade, para garantir a colaboração, uma boa colaboração. Então, até pelo que eu já falei aqui, testes de interface não cobre tudo, tem muitas coisas que são importantes e que você só pode avaliar, por exemplo esse negócio das métricas, ah quantas questões são fechadas, quantas questões são perguntadas você só pode avaliar com teste AB, no cenário real. Então assim, digamos assim que a aplicação pode ganhar um ponto no meu conceito, mas eu ainda não acho que é suficiente.

E. Quais são os principais riscos associados ao uso do G.A.M.E.? O G.A.M.E. ele não deixa de ser um game né? Um jogo, porque ele tem regras né? Eu tenho que seguir as regras, então eu tenho que criar minhas tarefas de colaboração e as issues baseadas nessas três dimensões. Então o risco que eu vejo é para eu me dar bem no G.A.M.E. eu vou levantar muitas issues pensando nessas 3 dimensões e talvez tenham outras dimensões aí que eu esteja ignorando. E o G.A.M.E. vai me dar talvez uma falsa sensação de segurança, porque eu estou bem no game, mas será que eu estou bem na vida real quando o sistema entrar em produção? O Gameplay Interaction aqui tem o player, e sempre você tem que escrever aqui quem é o player, e as vezes, não sei, na minha terceira issue aqui eu achei que o player não era uma pessoa ou um grupo de pessoas, mas a comunidade como um todo. No caso aqui eu queria dizer aqui que, ? eu tinha até escrito uma coisa diferente antes? não eu acho que o que eu escrevi antes, não? como o dono do site ou gerente de comunidade eu queria que as pessoas? ah eu até botei moderador, então acho que era mais ou menos isso, eu quero que as pessoas expliquem antes de fechar a questão. Então, não é que as pessoas elas querem explicar antes de fechar a questão, mas o dono do site que quer que o site prospere, ele quer que as pessoas expliquem, ou talvez poderia ser até a pessoa que fez a pergunta que quer que o outro explique, então são sempre 2 pessoas envolvidas, não é que uma pessoa quer fazer uma coisa, é que uma pessoa quer que a outra pessoa faça aquela coisa. Não acho que é o cenário mais comum, mas de qualquer forma é uma coisa que fica um pouco esquisita nesse template.

F. Como você atenuaria esses riscos? Poxa, é complicado! Porque a gente também não pode abrir deixar esse processo sem restrições porque aí ele perde o que ele tem de valor né? Tipo se eu coloco que tem essas e dimensões Cooperação, Comunicação e Coordenação, é bom porque me força a pensar nessas coisas e é ruim porque talvez eu fique acomodado e não pense em outras coisas, se é que existem essas outras coisas. Mas se eu tirar pode ser que eu pense só em um aspecto. Então não sei, quanto mais exemplos tiverem dessa abordagem sendo aplicada, melhor eu consigo ver como é que eu posso preencher esses campos e como é que eu posso até divergir um pouco do que está

aí. Por exemplo essa coisa do Player do Gameplay Interaction ter 2 players, isso não está explicito aqui como uma possibilidade do G.A.M.E. mas se tivessem exemplos que mostrassem isso sendo feito eu poderia também fazer isso.

G. Você usaria o G.A.M.E. em sua vida? Por que? Rapaz acho que sim. Porque ele tem essa coisa meio gráfica, compartimentalizada, porque eu acho que você colocar restrições é bom para a criatividade né? É bom pra, ajuda você a levantar problemas a pensar diferente. E é um negócio que pode ser usado até em grupo, várias pessoas preenchendo com post-it essa planilha. Eu estou falando isso porque eu já tinha ouvido falar no Business Canvas, assim essa semana eu fiz uma dinâmica lá, e me apresentaram o Project Model Canvas que é a mesma dinâmica do Business Canvas mas pra projeto, e eu vejo esse G.A.M.E. como se fosse um GAME? Canvas também né? Uma coisa que se fosse um documento texto assim seria mais difícil colaborar mas tendo esse formato bem definido e as pessoas podendo colar post-its nos lugares e discutir facilita muito a discussão né e a geração de ideias. E assim justamente por me parecer uma abordagem leve, com exceção do finalzinho, dos padrões.

H. Quais etapas do G.A.M.E. se encaixam perfeitamente em seu propósito? Por que? A proposta do G.A.M.E. ? bom é ajudar a fazer um sistema colaborativo que seja mais colaborativo, é promover a colaboração identificando problemas? eu estou me questionando agora se ele é um negócio que pode ser usado antes de ter um produto? Acho que sim né? Pensando bem aqui. Você tentaria prever os issues que aconteceria, eu acho que dá para fazer no nível de interface antes de implementar um sistema usar, então serviria também nessa etapa antes de ter um sistema pronto ou ter um sistema pronto e querer melhorar. Ou você quer resolver um sistema já que fomente a colaboração ou você quer aumentar a colaboração de um sistema que já existe. Acho que seria mais ou menos isso a proposta. E se o G.A.M.E. realmente ajuda nisso aí, quais etapas. Etapas, nessa etapa aqui tecnologia eu não vi grande influência assim, nesse caso aqui, talvez porque eu já assuma que as tecnologias são sempre a mesma coisa. A gente não tende a pensar em tecnologias muito diferentes de web, mobile. Eu estou olhando aqui por padrão, assim o custo-benefício certamente foi baixo do padrão aqui, eu estou tentando pensar aqui se de fato ele teve alguma influência. Por exemplo, esse primeiro aqui, o issue é que o botão lá a ação de perguntar uma pergunta está difícil de achar. Então assim, está claro para mim que então eu vou mudar de posição, mudar a cor, mudar a forma, mudar alguma coisa, né e isso eu estou pensando em termos de interface. E aí quando eu fui para o padrão, aí eu tive que reestruturar esse meu pensamento em termos de jogo e e em termo daqueles padrões lá. Então, o que era uma coisa simples, ah vou mudar a cor da posição ou vou mudar a fonte, virou um negócio de resource location, eu tive que aprender sobre vários jogos que usam localização de recursos para isso e para aquilo, então nesse caso aqui não ajudou não. No segundo, talvez, ajudou um pouco sim. Porque eu pensei nos dois jogadores, o que faz a pergunta e o que responde e quais são os objetivos deles, percebendo que eles têm objetivos comuns e aí eu até identifiquei um padrão lá que é Compartilhar Recompensas, que eu acho que não tinha pensado nisso antes de olhar o padrão, que se o cara ajuda a melhorar a pergunta ele deve ganhar pontos porque ele fez

isso. Nesse caso até ajudou, embora tenha dado muito trabalho para chegar até aí. Eu acho que ajudou, mas para o tempo que levou não sei se justifica. O resto é isso mesmo, se aplica bem a proposta. Os issues acho que foi bom pensar nessas issues, mas eu me deparei com aquele problema que eu pensei em coisas que acabei não usando lá na frente. Eu não sei se, porque aqui também eu peguei uma issue de cada, não se eu tentasse ser mais abrangente se eu chegaria a colocar essas issues aqui, de qualquer forma foi bom pensar nelas antes de pensar em qualquer coisa, até porque eu posso usar, trabalhar essas outras issues em outras abordagens que não envolvam testes de interface.

I. Quais etapas do G.A.M.E. não se encaixam perfeitamente em seu propósito? Por que? [Respondeu na pergunta anterior.]

C.3 TRANSCRIPTION OF INTERVIEW WITH SUBJECT 4

A. O G.A.M.E. correspondeu as suas expectativas e porquê? Sim, sim. Eu acho que assim no meu ponto de vista assim que não sou muito conhecido da área de engenharia de software e técnicas de desenvolvimento de software, sim tenho muito pouca experiência, eu acho que o contato com a essa, conhecimento né dessa ferramenta, framework eu acho que foi bem interessante porque eu acho que consegui aplicar um pouquinho desses, dessas técnicas de desenvolvimento e me ajudaram a perceber como é importante várias coisas de forma organizada. De você ter um? vamos chamar de metodologia né? Ter uma estratégia assim bem definida de como você vai organizar suas atividades, e como você vai executar e planejar e resolver o seu problema e implementar também. Eu acho que é bem interessante sim eu achei bem positivo nesse sentido. Para ajudar a guiar o que você vai fazer, desde a etapa de você planejar e você ter em mente o seu foco, não eu quero gamificar a Wikipedia, mas assim como eu vou fazer? Até de você pode organizar melhor suas ideias, se direcionar melhor né, ter um direcionamento melhor de como resolver o problema.

B. O que você gostou ao usar o G.A.M.E.? O ponto positivo no uso da ferramenta acho que é que justamente você conseguir ter essa metodologia bem clara assim de como seguir para aplicar gamification no software de interesse né? Acho que assim, você tem digamos, pelo menos o que eu percebi, você consegue extrair uns pontos positivos de algumas técnicas de desenvolvimento de software e trazer para sua ferramenta, e acho que isso é bem positivo. Eu tive uma experiência bem interessante assim, até assim para mim que não tenho familiaridade com essas técnicas de desenvolvimento de software me ajudou a me organizar mesmo no decorrer do processo para resolver o problema. E acho que deixar bem claro, deixar mais claro na minha cabeça quais são as etapas para desenvolver esse problema.

C. O que você não gostou ao usar o G.A.M.E.? O que eu não gostei assim, na verdade eu senti falta. Eu achei interessante a ideia de você gamificar uma ferramenta de crowd, de colaborativa que muitas pessoas usam. Mas o que eu senti falta foi de? talvez não esteja relacionada a ferramenta em si, mas um suporte para a ferramenta que me

pudesse ter mais exemplos. Porque eu não que não sou da área que não conheço muito gamification eu senti falta de ver como é que as pessoas aplicam isso na prática, mais exemplos para eu poder me situar melhor de como aproveitar ao máximo da ferramenta, do framework ou da técnica que você está propondo para eu poder digamos, ter noção melhor de como aplicar no software de interesse, por exemplo na Wikipedia. Como é que eu posso aplicar na Wikipedia. Como é que eu posso aplicar na Wikipedia aquilo ali. Eu conheço um pouquinho da Wikipedia, conheço um pouco do framework, da técnica, mas assim talvez um material de suporte, no meu caso que tenho pouco conhecimento me ajudasse a tomar algumas escolhas assim em que pontos da ferramenta, do software, da Wikipedia eu poderia aplicar gamification. Que pontos seriam mais interessante. Não sei se isso faz parte da metodologia. Talvez exemplos assim, digamos atrasou um pouco mais o meu desenvolvimento e talvez potencializasse se eu tivesse um materialzinho assim de suporte que pudesse mostrar, tem exemplo tal de como usar gamification assim, tem exemplo tal usou gamification assim, tal ferramenta usou gamification. Tu me falou do Waze, isso me ajudou muito quando tu falou do Waze, eu consegui entender bem, entender melhor, não sei bem né, entender bem melhor quando tu exemplificou o Waze e conseguiu explicar como o Waze usava o gamification. Isso me ajudou bastante para eu entender a ferramenta. Mas eu senti falta assim de outros exemplos porque aí eu poderia pensar nos exemplos e ver como é que aquele conhecimento que já foi aplicado nos exemplos poderia me ajudar a aplicar em uma nova solução. Aí foi talvez só uma dificuldade que eu tive.

D. O que você pensa sobre aplicações usando o G.A.M.E.? Assim eu acredito que com, criando um software com o G.A.M.E. o final seria que, como usuário final interessante seria que, o que parece interessante é a participação do usuário, então é que com essas, esses recursos que o framework traz, isso acho que pode possibilitar ao usuário a ter mais interesse pela ferramenta, a colaborar mais, a interagir mais, a enfim, aumentar a participação do usuário no uso da ferramenta. Talvez participação em termos de tempo que ele fica usando e talvez também em termos de tempo que ele fica usando e talvez também em termos de prazer de se sentir bem usando a ferramenta. Eu acho que eu esperaria isso. E acho que assim uma coisa de positiva, já falando conceitualmente é essa coisa de deixar o software mais divertido né? Que as pessoas interajam mais, possam interagir mais, acho que isso é uma ideia bem interessante. Como usuário acho que isso é muito positivo. Para toda comunidade, não só para o usuário individualmente, mas para toda comunidade. Acho que do ponto de vista do usuário final pode as vezes também a depender dos recursos que você use, pode ficar meio carregado né? O software tem que ser simples também então, tem que introduzir estratégias para atrair o usuário que torne o uso mais simples e mais fácil, mas que as vezes a gente coloca umas estratégias de uso que pode até atrapalhar né? O usuário pode até perder interesse pela ferramenta. Tem que ter muito cuidado com o que você coloca para atrair o usuário e tornar a ferramenta mais simples e mais fácil de usar. Acho que são esses cuidados mesmo. Às vezes você pode ter uma ferramenta que tá acostumado a usar, estava acostumado a usar a Wikipedia aí de repente alguém botou gamification lá aí a ferramenta ficou cheia de funcionalidades acabaram mais atrapalhando do que... é você saber dosar exatamente quais recursos

você pode trazer pro software de forma que ele atraia realmente e não afaste o usuário, acho que essa é a grande questão, exatamente como você atrair o usuário, aqui pode ser positivo, pode usar recursos que são interessantes mas pode afastar alguns usuários também né? Deixar a ferramenta mais trabalhosa. É o cuidado, você tem que fazer isso com muito cuidado. Você não pode fazer isso colocando qualquer coisa.

E. Quais são os principais riscos associados ao uso do G.A.M.E.? É.... de fazer a escolha certa né? De qual recurso você introduzir no software, na ferramenta. Eu acho que tem que ser um ponto muito relevante né, você tem que saber e tem que escolher direitinho e como projetar esse recurso para que atraia o usuário mesmo, então você tem que ter muito cuidado na hora de fazer essa escolha. Assim a ferramenta, a metodologia como um todo como você tem que tipo uma etapa de planejamento que você tem que pensar direitinho quais recursos vai ter que introduzir. Você tem também o, eu esqueci qual é o nome? então aquela etapa inicial ali de planejamento assim é importante e digamos diminui um pouco esse risco. Mas acho que esse risco já é inerente do processo de gamification, então você na hora de gamificar uma ferramenta colaborativa, eu acho que você tem que ter muito cuidado o que você vai inserir, e usando direitinho o framework você consegue digamos, diminuir um pouco esse risco. Você tem a etapa aqui de Gathering, né? Aí você realmente descreve direitinho objetivo e tal, funcionalidades, issues? então essa etapa inicial é extremamente importante para ficar claro o que você vai fazer, e então nessa etapa inicial você também é, refletir, e pensar e procurar, introduzir um recurso de gamification, que realmente traga a atenção do usuário, que chame o usuário para participar. Assim, no meu ponto de vista também de? usuário com pouca familiaridade com as técnicas de Engenharia de Software, por um lado e positivo que a gente consegue ver um processo né desse de Engenharia de Software funcionando. Ele é relativamente simples, mas também ele também ele é bem abrangente para atender todo processo de desenvolvimento, mas para mim que não sou muito engenheiro, não tenho muito conhecimento, acabo me confundido um pouco. Mas eu sei que assim de modo geral isso não é tão ruim. Você tem que se adaptar um pouquinho a alguns termos, a algumas metodologias da Engenharia de Software. Por um lado, é muito bom, mas para quem está de fora realmente vai ter que participar. Não que tenha que tirar, eu que a estrutura do framework tem que ser assim mesmo, mas para quem é pouco experiente vai ter esse carregado inicial para você se adaptar, familiarizar e utilizar. Mas entre as técnicas de Engenharia de Software que eu conheço, eu acho que você conseguiu aplicar muito bem, fazer as escolhas certas de Engenharia de Software para colocar aqui né? Das técnicas de desenvolvimento de software. Eu acho que as escolhas foram boas, apesar de não conhecer muito, mas dentro do universo que eu conheço as escolhas foram acertadas. Mas enfim, por se tratar de um processo, você tem que conhecer esse processo para aplicar bem. Acho que é isso.

F. Como você atenuaria esses riscos? Boa questão. Eu acho que, eu não sei como entraria também né, mas aquela ideia né de você ter um recurso em anexo em que você pudesse consultar casos de sucesso de gamification, algumas experiências passadas né? Não sei se isso se encaixa dentro do framework, mas no meu caso de usuário que

desenvolve, eu acho que é importante ter alguns casos de sucesso, conhecer assim, ter um histórico de sucesso para poder guiar as minhas escolhas de que recursos de gamification eu vou utilizar. Acho que é isso. Pronto e a questão da engenharia de software pode minimizar isso aí, é eu acho que assim talvez ter uma equipe, como essa coisa de gamification é uma coisa, é uma técnica que envolve vários aspectos né? Tanto de implementação, tanto de pensar na usabilidade de pensar na ferramenta como algo que seja atrativo e prazeroso, então eu acho que a equipe recomendada para usar o gamification, ou uma pessoa com essas várias habilidades, tanto de digamos de engenharia, de desenvolvimento de programação, com habilidade, com conhecimento de jogos também, acho que game design seria o termo né? Uma pessoa com essas 3 características ou uma equipe que pudesse ter esses diferentes perfis né de profissionais envolvidos na equipe. Acho que seria uma recomendação para aplicar o framework. Acredito que a habilidade é, você precisa realmente conhecer de engenharia de software, mesmo que seja pouco, mas é importante ter um conhecimento básico de desenvolvimento de software. É importante saber também de programação, mesmo que a programação seja menor, mas é importante ter uma noção de programação, já que está tratando de desenvolver essa ferramenta. E a questão de... eu chamo de game design, não sei se esse conceito está sendo aplicado correto. Acho que game design por que assim ele precisa tornar aquela ferramenta mais interativa, mais fácil de usar, mais divertida de usar, com recursos visuais que atraia, que chame a atenção do usuário para participar, para colaborar.

G. Você usaria o G.A.M.E. em sua vida? Por que? Assim, tem uma ferramenta que eu uso que eu acho bem interessante, é uma ferramenta colaborativa também e é uma rede social. E é uma ferramenta colaborativa para você colocar conteúdos sobre robótica, e acaba sendo uma rede social também. Acho que uma aplicação interessante do G.A.M.E. seria justamente dessa rede social, já existe uma comunidade por trás, já tem pessoas que postam, eu coloco conteúdo também, mas talvez pensar na ferramenta colaborativa com a estrutura de gamification seria mais atrativa né? Até porque tem muito usuário jovem, tem muitas oficinas, e aí o pessoal das oficinas traz a ferramenta para as oficinas e no final da oficina os instrutores estimulam os alunos a colocar o que foi gerado das oficinas nessa rede social. Então talvez em pensar como colocar o gamification essa rede social. E assim a experiência do dia a dia, acho que assim as técnicas que foram escolhidas, as metodologias de engenharia de software que foram escolhidas, acho que foi bem interessante e isso digamos acho que abriu um pouco a minha cabeça a pensar mais sobre digamos essas metodologias de desenvolvimento né? Que eu achei que foram interessantes e no futuro eu devo voltar a dar uma olhadinha nessas metodologias que você usou para também usar para mais para o meu cotidiano.

H. Quais etapas do G.A.M.E. se encaixam perfeitamente em seu propósito? Por que? Pronto, assim que o mais interessante assim que representa a metodologia, é a parte de gameplay, que você coloca aqui esses quatro atores, ou quatro características: player, action, state e feedback. Acho que isso aqui assim representa bem a sua ferramenta. Essa etapa de analise também dividida em cooperation, communication e coordination acho que representa bem o framework. A parte de execução eu acho que é mais

o resultado né, uma etapa final que precisa existir, seria importante né? Mas seria importante em muitos processos de desenvolvimento de Software. Mas uma característica mais peculiar acho que realmente essa etapa de Modeling, modelagem, que você tem aqui esse player, action, state e feedback e também essa etapa que tem cooperation tasks (cooperation, communication, coordination) acho que representa bem.

I. Quais etapas do G.A.M.E. não se encaixam perfeitamente em seu propósito? Por que? Talvez os menos relevantes seriam, assim eu não sei bem dizer. Mas assim a parte de resultados tal, aqui também tem que avaliar os resultados de implementação. Você olhar para o framework como uma metodologia para desenvolver o software. Essa etapa aqui de execução, mas de implementação, de resultado, ela talvez não seja tão relevante para a metodologia como um geral. Mas é importante ter uma forma de avaliar o resultado feito né, aí tem que existir também né, não que tenha que tirar, mas talvez de todo processo, talvez o que vejo como muito importante, é justamente você dar um norteamento para o desenvolvedor do que ele tem que fazer para atingir o gamification né, pra gamificar como uma ferramenta. E a etapa de implementação e resultados seria só digamos um plus, a gente está dando uma metodologia então é bom ver como ficou o resultado, o produto final, criado a partir dessa metodologia, mas assim do meu ponto de vista não seria tão relevante para caracterizar assim o framework.

C.4 TRANSCRIPTION OF INTERVIEW WITH SUBJECT 5

A. O G.A.M.E. correspondeu as suas expectativas e porquê? Rapaz, essa pergunta é difícil. Olha primeiro, ele correspondeu parcialmente tá? Apesar de ser uma proposta bem interessante, está bem definida as etapas e acho que estão claras as etapas e como elas se relacionam com as etapas anteriores e posteriores. Eu acho que esse, a etapa onde você se baseia naqueles padrões de game design, ela acaba sendo muito extensa e as vezes até redundante, o que eu quero dizer com isso. Eu acho que tem muitos padrões que se assemelham, o que acaba acontecendo, você acaba muito tempo lendo os padrões, padrões que se repetem e você acaba, se você tiver numa realidade de pouco tempo, você acaba prejudicando o tempo que você poderia se dedicar para as outras etapas, entendeu? Então talvez um subconjunto daqueles padrões pudesse ser mais eficiente entendeu? Com relação aos testes de usabilidade, são 4 tipos de testes que a ferramenta que você usou disponibiliza, que são: o five test, o click test, o navigation test e o preference test. Sinceramente eu não achei que nenhum dos quatro se adequava a avaliação das alterações que eu tinha feito entendeu? Porque eu acho, não sei, eu imaginava que algum dos testes iria fazer uma comparação entre as duas soluções e ver se houve melhora em algum quesito em relação a solução em relação a outro. Mas os testes se afeta dessa forma. Outra coisa que também não ficou muito clara foi essa questão do state, eu acho que ela acabou se tornando muito repetitivo, basicamente, o lugar da interface que você está. Eu não consegui estabelecer muito bem um valor para esse aspecto aí da modelagem, o state.

B. O que você gostou ao usar o G.A.M.E.? Cara, o que eu gostei é que uma forma sistematizada, bem clara de aplicar gamificação em projetos, no caso digitais mas

poderia ser aplicada a outros tipos de projeto. É uma forma bem clara. E fornece o processo todo né? Você forneceu o processo todo. Desde o início que você só faz a analise até você fazer o teste de usabilidade lá. Então, é uma forma bem clara e completa né? Bem sistemática, sistematizada e é claro completa.

C. O que você não gostou ao usar o G.A.M.E.? O que eu não gostei foi o que eu falei na questão de expectativa né? A questão do state eu tive dificuldade em entender a importância desse aspecto na modelagem, eu acho que pelo menos para um projeto digital de aplicativo, aplicações em geral. Eu não sei se isso poderia ser revisto, considerando outro aspecto, não sei. E a questão dos testes, eu não soube bem como escolher o teste, na verdade eu não achei que nenhum teste era melhor, era tão adequado para avaliar o que eu fiz. Eu pensei, tanto que eu pensei que todos poderiam ser usados, não se destacou com uma abordagem de teste que era mais pertinente entendeu? Eu executei o 3.2 antes do 3.1 pois eu achei que sabendo a forma de interação e qual é a ação e o feedback eu poderia escolher um padrão de game design melhor. Com relação aos padrões foi o que eu falei, eu achei a lista muito extensa e ela é um pouco redundante. Então talvez, usar uma lista mais enxuta, juntar alguns daqueles padrões ali poderia ajudar, na hora de estabelecer esses padrões entendeu.

D. O que você pensa sobre aplicações usando o G.A.M.E.? Eu esperaria basicamente duas coisas, duas coisas me vêm à cabeça: uma é que essa aplicação ela envolva elementos de gamificação e dois é que ela tenha melhor usabilidade do que uma aplicação que não usa entendeu? Porque acho que a metodologia me convenceu de que ela é capaz de tornar uma aplicação melhor em termo de usabilidade. Deixa eu ver se teria mais alguma coisa que eu poderia pensar?

E. Quais são os principais riscos associados ao uso do G.A.M.E.? Quais são os riscos que eu acho que exista. Olha, uma coisa que até eu fui vítima e a maioria das pessoas podem ser é?fazer um uso meio condicionado, entendeu? É que já existem várias aplicações gamificadas e quem trabalha com gamificação já está acostumado com certas formas de gamificar. Então você acaba de alguma certa forma sendo direcionado a usado essas mesmas formas de gamificação. Quando eu lhe falei que eu lhe falei que a lista de game design extensa eu até fiquei me perguntando: será que está muito extensa ou será que eu desconsiderei padrões de game design que para mim não usados, que na minha experiência não são usados em gamificação e por isso eu estou condicionado e aí eu achei que não seriam relevantes ou seriam redundantes. Então qualquer pessoa que use vai estar condicionada a isso. Resumindo, o risco é usar os padrões que elas já estão acostumadas em ver em outras aplicações gamificadas. Então o benefício que você teria com o G.A.M.E. que poderia ser diversificar esses tipos de padrão, você acaba perdendo porque a pessoa já vai de forma direcionada, talvez até antes de começar a usar o G.A.M.E. ela já imagina quais são os padrões que ela vai usar naquela aplicação que ela escolheu. E ela vai direcionar toda a modelagem dela para usar aqueles padrões. Talvez usar o teste de usabilidade errado, como eu falei, eu tive dificuldade de escolher o teste

de usabilidade. Talvez a pessoa corra o risco de escolher também um teste que não é o mais adequado para a alteração que ela fez e aí acabar tendo resultados que não são reais. Não são verdadeiros.

F. Como você atenuaria esses riscos? O risco da questão do teste de usabilidade, eu buscaria fornecer para o usuário instruções de como escolher o teste, o usuário do framework, usuário da metodologia. Algo tipo, esse five seconds test é mais pertinente para quais tipos de mudança, ou o click test é mais pertinente para quais tipos de mudanças. E para evitar o condicionamento em relação a experiência que já tem? isso é até complicado cara. Isso é complicado. Sei como resolver isso não. Não sei responder agora.

G. Você usaria o G.A.M.E. em sua vida? Por que? Rapaz, eu usaria sim em meu ambiente profissional, eu acho que usaria como eu falei pelos aspectos positivos que ele tem. Uma coisa que eu achei interessante também que eu não mencionei antes, é que eu achei que pela estruturação da metodologia estar bem coesa, o processo de gamificação fica até mais rápido do que quando você começa a ter ideias uma forma adhoc, sem nenhuma metodologia. E além disso ele já direciona você para fazer os testes de usabilidade e para analisar resultados. Ela não te leva só até a modelagem, então você é capaz de avaliar aquele resultado, se aquela sua alteração chegou de fato funcionou, gerou algum resultado. Eu não conheço muitas metodologias de gamificação. Eu conheço alguns direcionamentos daquele curso do Coursera né? E eu não vi nenhum tipo de metodologia que fosse tão abrangente e ao mesmo tempo tão enxuta e prática, entendeu? Achei que ficou bem prático, como eu falei, as etapas bem coesas e bem relacionadas, então por isso eu usaria, usaria na minha vida profissional sim.

H. Quais etapas do G.A.M.E. se encaixam perfeitamente em seu propósito? Por que? Olha, eu acho que todas as etapas com exceção das etapas 1.3 e 2.1.1 foram relevantes no meu projeto. Não sei se seriam relevantes em outro, mas não foram relevantes no projeto que eu fiz. Sequer foram consultadas no restante das etapas. Bem, primeiro, a etapa de objetivo: você dividiu a etapa em objetivos individuais e objetivos comunidade. Isso é interessante porque você precisa fazer o seu projeto, sua melhoria o que quer que seja, ou uma criação de grupo, considerando esses 2 aspectos, essa informação é básica né, quando você for analisar problemas relacionados, enfim, as atividades cooperativas, problemas relacionados, você vai precisar dessas duas informações, são informações básicas. Você vai precisar fazer o seu produto considerando o que um usuário vai querer, mas também o que a comunidade precisa que o usuário em conjunto forneça. As funcionalidades elas são importantes, acredito que seja mais importante para você ter uma noção de quais são as ferramentas que o seu produto fornece e como você pode melhorar essa ferramenta, esses recursos, então você define, escreve o que tem de recurso na sua ferramenta. Que podem resolver seu produto. Problemas, acho que da etapa um é uma das coisas mais importantes. Você precisa exatamente direcionar o seu projeto para solucionar os problemas que você identificou na ferramenta. A etapa 2.1.2 está muito relacionada a etapa 1.2, então os recursos que a ferramenta oferece vão per-

mitir que o usuário faça essas ações aí. Então tudo que foi mencionado para a etapa 1.2 acho que se aplica a etapa 2.1.2 também. Bom, na etapa 2.2, você basicamente pega o que está na 1.2 e na 2.1.2 na classifica né de acordo com Cooperação, Comunicação e Coordenação. E você logo em seguida associa os problemas relacionados a essas atividades, que também é algo muito, como se você pegasse as informações anteriores e classificasse. Você já relacionou os problemas em 1.4. E você agora está classificando para poder atacar melhor os problemas individualmente. Então também são peças bem importantes e bem estruturadas. Você modela, quer dizer eu no caso modelei a interação do gameplay, uma vez que eu já sei quais são os problemas e quais são os objetivos. Eu modelei o que o usuário queria enquanto jogador e depois identifiquei os padrões que poderiam ser aplicados para selecionar aqueles padrões que eu modelei, então é uma das etapas mais significativas. Planejamento você escolhe um teste de usabilidade para poder executar e avaliar o seu resultado, acho que o 4.1 e o 4.2 são bem ligados, são duas etapas muito ligadas. Me confundiria até, também poderia ser uma etapa só. A etapa 4.3 você apresenta suas soluções e os resultados também, então, enfim.

I. Quais etapas do G.A.M.E. não se encaixam perfeitamente em seu propósito? Por que? Como eu falei, acho que as etapas menos significativas para minha execução foi a 1.3 e a 2.1.1. A 1.2 é porque a tecnologia em nenhum momento influenciou as etapas posteriores. Eu não sei se eu entendi errado, ou se é comum, mas para mim essa informação não foi usada em momento nenhum, após eu ter definido isso aqui. E a 2.1.1 também não, porque ela ficou implícita, eu achei redundante, ela estava implícita nas funcionalidades e no que os usuários fazem. Não sei, acho que a 2.1.1 eu também não utilizei mais ela, a partir do momento que eu passei dessa etapa aí, eu não usei mais essa informação. Todas as informações, com exceção da 1.3 e da 2.1.1 eu utilizei novamente. Me ocorreu agora também cara, que eu executei a etapa 4.3 antes da 4.1 e da 4.2. Eu também troquei essa ordem. Eu defini como seria a solução antes de definir como seria a avaliação. Porque eu imaginei que o teste de usabilidade que você iria aplicar estaria muito relacionado com a mudança que você fez no produto, entendeu? Se você fizer uma mudança de tal tipo, a avaliação, por exemplo? se for você mudança na navegação, você iria fazer uma avaliação de um navigation, você iria fazer um navigation test, entendeu? Se você fez uma mudança é, por exemplo...de...acrescentou uma nova informação, ou mudou a forma como a informação é exibida você iria fazer um five second test ou um preference test, então eu imaginei que para definir qual teste seria interessante que eu definisse a mudança que eu ia fazer. Essa foi a minha lógica. Não só definir o que seria feito. Eu estava falando que para definir o teste de usabilidade que seria usado seria melhor saber quais mudanças seriam feitas, mas não só saber quais mudanças serias feitas, mas visualiza-las também, entendeu? Dessa forma 4.3 aí eu descrevi isso, mas eu também já fiz os esboços visuais lá, os protótipos. Então visualizando essas mudanças eu achei que teria uma percepção melhor de qual teste de usabilidade usar. [Sobre trocar a ordem do 3.2 e 3.1]Me pareceu mais lógico isso, eu primeiro saber o que é que o usuário espera né? A estória dele meio que diz o que ele é que ele espera, o que ele faz, o que ele é. Então, entender o que ele espera, qual é a ação que ele vai executar e qual o feedback que ele quer ter, para depois definir qual padrão que eu iria usar para contemplar aquela estória que

eu defini entendeu? Eu acho que se eu definir o padrão antes, eu não consigo visualizar, eu realmente não consigo visualizar, definir esse padrão antes da estória, porque eu acho que a estória vai ficar condicionada ao padrão que você definiu. Assim, a estória fica restringindo o conjunto de padrões que você pode usar, mas acho que é uma restrição menor. Ou talvez nem restrinja, talvez essa questão de a gente estar acostumado a usar os mesmos padrões para solucionar os problemas de gamificação, já está habituado a fazer isso, esteja direcionando esse aspecto. Como eu acho que aconteceu, eu acho que a etapa crítica está aqui, que para mim foi 3.2 e depois 3.1 para não dizer ao contrário, é quando eu defini a estória e fui passar para o padrão eu já fui meio direcionado, mas não porque eu defini a estória primeiro, e sim porque eu já estava habituado a outras soluções de gamificação que usam padrões bem similares. Eu acho que a questão de tempo também acaba definindo isso, a gente o dia a dia na correria, a gente quer terminar rápido as coisas, se um requisito dessa aplicação, dessa solução aí fosse criatividade e tivesse mais tempo para trabalhar nela, eu talvez eu teria, eu tivesse utilizado outros padrões, talvez eu tivesse sido mais ousado em relação as soluções dos padrões de gamificação utilizados.

Appendix D

This appendix presents the extraction and the coding of the qualitative evaluation with software developers. The data was extracted from the interviews presented in Appendix C

EXTRACTION AND CODING OF TRANSCRIPTIONS

D.1 EXTRACTION AND CODING OF INTERVIEW WITH PARTICIPANT 2

A. O G.A.M.E. correspondeu as suas expectativas e porquê?
Extraction:

1 - Correspondeu as expectativas, porque a forma como que ele é feito, ele tem etapas bem definidas, então acho que eu esperava era justamente isso.

2 - Um guia para dizer as coisas básicas a cumprir para chegar em um produto final, e o G.A.M.E, faz exatamente isso.

3 - O G.A.M.E. Diz o que buscar no começo, o que analisar, como pensar em resolver e como testar.

4 - O G.A.M.E. funcionou legal para o que esperava.

Coding:

2A1 - Correspondeu as expectativas (1, 4)

2A2 - Tem etapas bem definidas (1).

2A3 - É um guia para dizer as coisas básicas a cumprir para chegar num produto final (2).

2A4 - Diz, o que pensar no começo, o que analisar, como pensar em resolver e como testar (3).

B. O que você gostou ao usar o G.A.M.E.?
Extraction:

1 - Gostei da etapa de Modelling pois as estórias ajudam basdtante a ver o que você precisa e como é que vai chegar nisso;

2 - É interessante ler uma estória com sujeito, problema e o que ele espera.

3 - Gostei de na parte da Execução a distinção do planejamento, instrução e implementação pois dá uma clareza melhor de como aquilo vai funcionar.

4 - Tentar entender como fazer o que eu tinha pensado e colocar no teste ajudou a clarear as ideias, emtre fazer a instrução e dizer como foi implementado.

5 - Modeling e Execução foram as coisas que achei mais bacanas, pois nunca tinha feito uma sequência ocmpleta assim.

6 - Saber o que eu preciso, planejar, criar as instruções do que vai ser executado, implementar e testar. Acho essa sequência muito legal.

7 - A sequência de tudo é interessante mas a parte de Gathering e Analysis não é tão clara.

8 - Quando você parte pro Modeling e Execution você começa a entender mais aquilo que você queria com Analysis e o Gathering.

9 - A ideia não está muito clara no começo, e quando você passa pra parte da estória e do planejamento das instruções você começa a clarear a sua própria ideia.

10 - As etapas como estão feitas, principalmente em Modeling e Execution acho que ficou muito fluido.

Coding:

2B1 - Gostou mais das etapas de Modelling e Execução pois estão mais fluidas(1,3,5,10);

2B2 - Interessante ler uma estória com sujeito, problema e o que ele espera (2)

2B3 - Etapa de Execução dividida em planejamento, instrução e implementação dá uma clareza melhor de como vai funcionar (3,4);

2B4 - A sequência de tudo é interessante mas a parte de Gathering e Analysis não é tão clara (7).

2B5 - No Modeling e Execution você começa a entender mais o que você queria com Analysis e o Gatheering (8,9);

C. O que você não gostou ao usar o G.A.M.E.?
Extraction:

1 - Não gostei muito da parte de Patterns

2 - Patterns são muito úteis mas são muito complexos;

3 - O G.A.M.E. fica muito dependente de você saber os os padrões;

4 - As coisas que tive mais dificuldade foi em dizer o que era cada elemento, o que é uma issue, tecnologia...

5 - A explicação das coisas não estava muito clara as vezes;

6 - Dividir em Cooperação, Comunicação e Coordenação ficou confuso pois tem coisas que se aplicam a mais de um;

Coding:

2C1 - Não gostou dos Padrões pois são muito complexos (1,2);

2C2 - O G.A.M.E. depende de saber os Padrões (3);

2C3 - Dificuldade em identificar o que era cada elemento pois a explicação não estava clara (4,5);

2C4 - Categorizar em Cooperação, Comunicação e Coordenação é confuso pois algumas tarefas se aplicam a mais de um;

D. O que você pensa sobre aplicações usando o G.A.M.E.?
Extraction:

1 - Eu espero que as funcionalidades que as aplicações tentam implementar e que tenham a ver com o gameplay sejam mais integradas ao sistema ao invés de adições para parecer que é um jogo.

2 - Espero que por ser uma coisa que foi usado o G.A.M.E. na sua concepção, que o gameplay e a gamificação seja mais integrada ao sistema, e não só adicionando coisas em lugares que funcionaria colocar ponto, score ou badge.

Coding:

2D1 - As aplicações tenham o gameplay integrado ao sistema ao invés de adicionar pontos, scores e badges (1,2);

E. Quais são os principais riscos associados ao uso do G.A.M.E.?
Extraction:

1 - Talvez as pessoas tentem colocar funções ou usar patterns que não caibam com o tipo de aplicação que está desenvolvendo.

2 - Patterns são legais porque é uma coisa que tem uma descrição de como funcionam e em teoria foram testadas, mas pode te desvirtuar com algo que não seja necessário na aplicação.

3 - Tem alguns tipos de patterns que você pode criar com o G.A.M.E., que talvez você não consiga visualizar que você não possa entender.

4 - Se não souber usar a pattern você acaba se perdendo.

5 - Padrão tem muita variavel, muita coisa, e uma coisa leva a outra - é muito complexo.

Coding:

2E1 - Patterns são complexos, muitas variaveis envolvidas, dificeis de visualizar e podem fazr o desenvolvedor se desvirtuar usando patterns que não casam com a aplicação (1, 2, 3, 4, 5).

F. Como você atenuaria esses riscos?
Extraction:

1 - Diminuir o nível de complexidade das possibilidades dependendo do tipo de aplicação.

2 - Diminuir o nível de patterns acessiveis para um tipo de aplicação.

3 - Existem funções que são muito especificas de certos generos de jogos, assim como existem funções especificas de certos tipos de aplicações.

4 - Fazer categorias de padrões mais usaveis em um tipo de aplicação facilitaria o entendimento e a combinação de coisas para não se perder num mundo imenso.

Coding:

2F1 - Reduzir e categorizar os patterns de acordo com os tipos de aplicações (1, 2, 3, 4).

G. Você usaria o G.A.M.E. em sua vida? Por que?
Extraction:

1 - Eu acho que provavelmente sim.

2 - Acho que é o G.A.M.E. é uma ferramenta bem suscinta, bem objetiva, tem etapas muito bem definidas, e não são etapas que levam muito tempo para serem feitas.

3 - As coisas que tive mais dificuldade eu levei menos de 40min pra fazer.

4 - Se vocÊ tem uma coisa que precise ser resolvido com agilidade, é uma ferramenta interessante de se tentar usar, porque o fator tempo de produção dentro do G.A.M.E. é uma coisa muito interessante.

5 - Coisas que se eu não tivesse um método como o G.A.M.E. eu levaria muito mais tempo pra pensar e acho que levei menos tmepo tendo pontos idnividuais e objetivos para repsonder.

6 - Deu agilidade na construção das minhas ideias e na execução.

Coding:

2G1 - Provavelmente usaria o G.A.M.E (1).

2G2 - G.A.M.E. é uma ferramenta suscinta, objetiva com etapas em definidas e que rápidas de serem executadas (2,3,4);

2G3 - Com o G.A.M.E. em pouco tempo resolveu pontos individuais e objetivos, pois agilizou na construção e execução das ideias (5,6);

H. Quais etapas do G.A.M.E. se encaixam perfeitamente em seu propósito? Por que?

Extraction:

1 - Acho que é importante separar a parte dos objetivos em individual e comunitário, porém as vezes podem ser muito iguais;

2 - O objetivo acho que o lugar que ele tá é importante, porque antes de resolver o problema eu preciso saber com o que eu estou lidando.

3 - A primeira coisa que você tem que saber é, o que é, que plataforma é essa e o que você está fazendo, o que a plataforma pretende e o que ela está trazendo pros usuários, como objetivo individual e da comunidade.

4 - Acho o objetivo é muito importante estar no começo porque ue preciso entender o que é que eu estou trabalhando pra tentar resolver os problemas.

5 - A separação do Modeling em estórias pras categorias foi bem colocada no processo porque depois se colocar como usuário tentando resolver o problema antes de tentar planejar e executar é uma forma interessante de ter perspectiva naquilo que está fazendo.

Coding:

2H1 - Importante dividir os objetivos em individual e comunitário podem as vezes podem ser idênticos (1);

2H2 - O objetivo precisa estar no começo para identificar o que a aplicação é, o que faz, o que pretende e o que traz pros usuários (2,3,4);

2H3 - No Modeling, o uso de estórias pra cada categoria foi interessante para visualizar o problema como um usuário (5);

I. Quais etapas do G.A.M.E. não se encaixam perfeitamente em seu propósito? Por que?

Extraction:

1 - Tecnologia é importante pra pensar na aplicação das coisas, mas talvez não seja tão importante pra tentar entender uma função de resolver o problema.

2 - Tecnologia não seria no Gathering mas em uma pós-produção, pelo menos pra mim foi assim.

3 - Tecnologia talvez não tenha se encaixado no lugar que ela está.

4 - Issues, crowdsourcing e tasks, ficaram repetitivas devido a listar e depois separar.

5 - Talvez tentar colcoar essas classificações já essas issues por cateogria, fique claro pra entender quais são os problemas e onde eles estão colocados.

6 - Implementing e results, eu não sei se o Usability Hub atendeu ao nivel de interferencia que eu tentei fazer no sistema.

7 - Tive dificuldade em conseguir adequar o que eu precisava passar pra ferramenta pro teste.

8 - Talvez tenha sido um problema do UsabilityHub, talvez não atenda a todas as necessidades do que eu projetei ou eu não soube colocar direito.

Coding:

2I1 - Tencologia é importante pra pensar na aplicação das coisas, mas não é importante pra entender o problema - não se encaixou no Gathering (1,2,3);

2I2 - Issues, crowdsourcing e tasks ficaram repetitivas devido a identificar e depois separar (4,5);

2I3 - O UsabilityHub não atende para testar as mudanças necessárias (6,7,8);

D.2 EXTRACTING AND CODING OF INTERVIEW WITH PARTICIPANT 3

A. O G.A.M.E. correspondeu as suas expectativas e porquê?
Extraction:

1 - Acha que foi de acordo com o que estava esperando.

2 - Acho que foquei que me ajude com boas práticas, que me oriente a fazer as coisas no caminho certo, e essa orientação teve sim.

3 - Princiaplmente quando você divide em Cooperação, Comunicação e Coordenação.

4 - Me faz pensar em coisas que talvez eu não pensaria e que são importantes.

5 - Minha expectativa era que me ajudasse com boas práticas a fazer coisas que fossem boas práticas, e acho que o G.A.M.E. ajudou principalmente quando dividiu em Cooperação, Comunicação e Coordenação, que fez obrigatoriamente a pensar em aspectos que talvez eu não pensasse se fizesse sem isso.

6 - O Gathering foi importante para refletir sobre o papel do site, o que se espera obter com o site, e o que as pessoas fazem.

7 - A parte do padrão foi a parte mais demarada e foi uma experiencia diferente, inusitada.

8 - Demorou para se acostumar com o padrão, pois são falados em termo de jogador e de jogo e aqui eu penso na colaboração.

9 - O que ficou mais diferente é que no jogo tem o espaço do jogo, o mundo do jogo, e aqui uma interface gráfica.

10 - Algumas coisas não ficaram um mapeamento tão claro, mas ao mesmo tempo ver sob essa perspectiva de jogo faz você pensar em coisas diferentes também, faz você ver a colaboração sob uma outra ótica, que eu ahco que é bom para a criatividade.

11 - Execução, o que eu percebi foi que algumas coisas não conseguiria testar usando teste de interface, algumas coisas você tem que ver o site rodando pra ver o resultado.

12 - Não consegui executar o teste para a coordenação.

13 - O Gameplay Interaction Model ajudou, o player e o action são bem diretos da issue, e o state ajuda onde você trataria isso, e o feedback ajudou a pensar qual o objetivo disso e se está de acordo com os objetivos que eu listei no inicio.

14 - O Gameplay Interaction Model me ajuda a ver como eu vou medir, tendo bem claramente qual é o objetivo, eu sei como é que eu ovu medir, tanto é que o último objetivo, o terceiro lá eu vi que era algo que eu não iria conseguir medir.

Coding:

3A1 - Correspondeu as expectativas (1,5);

3A2 - Ajudou a executar boas práticas orientando a fazer as coisas no caminho certo (2,5);

3A3 - Dividir as atividades em 3C fez pensar em coisas que não pensaria e são importantes (3,4,5);

3A4 - Gathering foi importante para refletir sobre o papel do site, o que se espera obter com o site e o que as pessoas fazem (6);

3A5 - A parte do padrão foi a parte mais demorada (7,8);

3A6 - Padrões são pensados em termo de jogo e jogador e não de colaboração (8);

3A7 - No jogo tem o espaço do jogo, no site uma interface gráfica (9);

3A8 - O mapeamento de algumas coisas de jogo em um site não ficaram tão claras, porém ver sob essa perspectiva de jogo faz pensar em coisas diferentes, ver a colaboração sb outra ótica, bom para a criatividade (10);

3A9 - Na Execução, algumas coisas não conseguiria testar com teste de interface (11,12);

3A10 - No Gameplay Interaction Model, o state ajuda a ver onde tratar a issue (13);

3A11 - No Gameplay Interaction Model, o feedback ajuda a pensar qual o objetivo da estória e se está de acordo com os objetivos iniciais; (13)

3A12 - O feedback ajuda a ver como medir, e antecipar algo que não é possível de medir (14);

B. O que você gostou ao usar o G.A.M.E.?
Extraction:

1 - Gostou da sequência de passos que ajudou a pensar desde o inicio em coisas úteis;

2 - Gathering foi util pra Analysis, Analysis foi util pra Modelagem;

3 - Eu como programador quero ir logo pra Execução, e o G.A.M.E. me induziu a não fazer isso e isso afetou positivamente o resultado;

4 - Essa coisa dos 3Cs acho que foi legal também;

Coding:

3B1 - Sequência de passos me induziu a não ir direto para a Execução, e a pensar em coisas úteis desde o Gathering que ajudou a Analysis que ajudou a Modelagem (1,2,3);

3B2 - Categorizar com o 3C foi interessante (4);

C. O que você não gostou ao usar o G.A.M.E.?
Extraction:

1 - Demorei muito tempo nos Patterns e não gosto de demorar muito tempo em alguma coisa;

2 - Se os Patterns pudessem ser mais destrinchados ou tivesse alguma orientação, acho que fiquei muito solto;

3 - Poderia quebrar o Patterns em várias etapas menores pra ter um feedback rápido de que estou avançando no processo;

4 - Se tivesse alguma orientação, uma organização dos Patterns pensada no cenário de sistema colaborativo, ficaria mais fácil;

5 - Foi maçante essa tarefa de achar os padrões, pois nas etapas de Analise e Modelagem eu tentava achar problemas que conseguiria executar, mas acabei caindo em um problema que não consegui executar;

Coding:

3C1 - A etapa dos Patterns foi maçante e levou muito tempo e acabou caindo em um problema que não foi possível executar (1,5);

3C2 - Patterns poderiam ser destrinchados em etapas metores pra ter um feedback rápido do avanço no processo (2,3);

3C3 - Patterns poderia ter uma orientação, organização pensada no cenário de sistemas colaborativos (2,4);

D. O que você pensa sobre aplicações usando o G.A.M.E.?
Extraction:

1 - Um produto desenvolvido com o G.A.M.E. pensa em Cooperação, Comunicação e Coordenação, portanto tem uma visual mais holistica da colaboração e menos behaviourista.

2 - Não estou seguro que o 3C e o uso de testes de interfaces são suficientes para garantir qualidade na colaboração;

3 - Testes de interface não cobrem tudo, tem coisas que são importante que só são possiveis de avaliar na aplicação;

4 - A aplicação ganha um ponto no meu conceito mas não é o suficiente;

Coding:

3D1 - Um produto desenvolvido com o G.A.M.E. pensa em Cooperação, Comunicação e Coordenação, portanto tem uma visual mais holistica da colaboração e menos behaviourista (1).

3D2 - Não estou seguro que o 3C e o uso de testes de interfaces são suficientes para garantir qualidade na colaboração, pois testes de interface não cobrem tudo e tem coisas que são importante que só são possiveis de avaliar na aplicação real (2,3,4);

E. Quais são os principais riscos associados ao uso do G.A.M.E.?
Extraction:

1 - O G.A.M.E. não deixa de ser um jogo porque tem regras que tem que ser seguidas como criar as atefas de colaboração e issues baseadas no 3C;

2 - O risco é de pra me dar bem no G.A.M.E. eu vou levantar muitas issues pensando nas 3 dimensões (3C) e talvez existam outras dimensões que eu esteja ignorando;

3 - O G.A.M.E. talvez dê uma falsa sensação de segurança, porque estnado bem no game, será que estarei bem na vida real quando o sistema entrar em produção;

4 - O Gameplay Interaction Model, tem o player, e sempre vocÊ tem que escrever quem é o player, e as vezes o player não é uma pessoa ou um grupo, mas a comunidade como um todo.

Coding:

3E1 - O G.A.M.E. é um jogo pois tem regras definidas que tem que ser seguidas como por exmeplo, criar tarefas de colaboração baseadas no 3C pode dar uma falsa sensação de segurança: será que indo bem no game estou bem na vida real (1,3);

3E2- O risco é que se dar bem no G.A.M.E. levantando issues pensandas nas 3 dimensões (3C) e ignorar outras dimensões (2);

3E3 - No Gameplay Interaction Model, o player as vezes não é uma pessoa ou um grupo, mas se uma comunidade (4);

F. Como você atenuaria esses riscos?
Extraction:

1 - Deixar o processo sem restrições ele perde o que tem de valor;

2 - O 3C força a pensar nessas coisas e é ruim porque talvez fique acomodado e não pense em outras, se é que existem outras coisas;

3 - Se tirar o 3C, talvez eu só pense em um aspecto;

4 - Quanto mais exemplos tiverem dessa abordagem sendo aplicada, melhor eu consigo ver como é que eu posso preencher esses campos e como é que eu posso até divergir um pouco do que tá aí.

Coding:

3F1 - Complicado, pois tirar as restrições perderia o que ele tem de valor, ex. tirar o 3C poderia só pensar em um aspecto (1,2,3);

3F2 - Exemplos de uso da abordagem sendo aplicada melhor para saber como preencher os campos ou até divergir um pouco (4);

G. Você usaria o G.A.M.E. em sua vida? Por que?
Extraction:

1 - Acho que sim, pois ele tem essa coisa gráfica, compartimentalizada e eu acho que restirções ajuda a criatividade;

2 - Ajuda a levantar problemas e pensar diferente;

3 - Pode ser usado em grupo, varias pessoas preenchendo post-its nessa planilha;

4 - O G.A.M.E. lembra o Business Canvas;

5 - Se fosse um documento texto, seria mais dificil de colaborar, mas tendo esse formato bem definido e as pessoas pode colar post-its e discutir facilita a discussão e geração de ideias.

6 - Parece ser uma abordagem leve, com exceção dos patterns;

Coding:

3G1 - Acho que sim pois é uma abordagem gráfica e compartimentalizada com restrições que ajuda a criatividade, a levantar problemas e a pensar diferente (1,2);

3G2 - Pode ser usado em grupo com várias pessoas preenchendo post-its como o Business Canvas, por ter um formato bem definidido ajuda a discussão e geração de idéias (3,4,5);

3G3 - Parece ser uma abordagem leve, com exceção dos Patterns;

H. Quais etapas do G.A.M.E. se encaixam perfeitamente em seu propósito? Por que?

Extraction:

1 - Etapa de tecnologia não vi grande infuência, talvez porque eu já assuma que as tecnologias são sempre as mesmas (web, mobile);

2 - Patterns, o custo beneficio certamente foi baixo: ajudou mas deu muito trabalho;

3 - Os demais se aplicam bem as proposta;

4 - Foi bom pensar nas issues, mas me deparei com um problema das coisas que pensei não conseguindo usar lá na frente.

Coding:

3H1 - Com exceção de tecnologia e patterns, as etapas se aplicam bem na proposta (3)

I. Quais etapas do G.A.M.E. não se encaixam perfeitamente em seu propósito? Por que?

Extraction:

[Mesma extração do item anterior.]

Coding:

3I1 - Tecnologia não teve influência, talvez por assumir que as tecnologias são sempre as mesmas (1); 3I2 - Patterns ajudaram mas deu muito trabalho - baixo custo-beneficio (2);

3I3 - Foi bom pensar em issues mas ocasiou uma situação que não foi possível executar (4);

D.3 EXTRACTING AND CODING OF INTERVIEW WITH PARTICIPANT 4

A. O G.A.M.E. correspondeu as suas expectativas e porquê?

Extraction:

1 - Correspondeu as expectativas.

2 - Uma estratégia bem definida de como organizar as atividades, executar, planejar, resolver e implementar.

3 - Ajuda a guiar o que fazer, desde a etapa de planejar tendo em mente o foco até poder organizar melhor as idéias, se direcionar, ter um direcionamento de como resolver o problema.

Coding:

4A1 - Correspondeu as expectativas (1).

4A2 - Uma estratégia bem definida de como organizar as atividades (2).

4A3 - Ajuda a guiar o que fazer, desde a etapa de planejar tendo em mente o foco até poder organizar melhor as idéias e ter um direcionamento de como resolver o problema (3).

B. O que você gostou ao usar o G.A.M.E.?
Extraction:

1 - O ponto positivo é ter essa metodologia bem clara de como seguir para aplicat gamification no software de interesse

2 - Conseguiu extrair pontos positivos de algumas tecnicas de desenvolvimento de software e trazer pra sua ferramenta

3 - Me ajudou a organizar no decorrer do processo de resolver o problema

4 - Deixou mais claro quais são as etapas pra desenvolver esse problema;

Coding:

4B1 - Uma metodologia clara com pontos positivos de tecnicas de desenvolvimento de software pra aplicar gamification em um software de interesse (1,2);

4B2 - Ajuda a organizar o processo de resolver o problema (3,4);

C. O que você não gostou ao usar o G.A.M.E.?
Extraction:

1 - Senti falta de um suporte pra ferramenta que pudesse ter mais exemplos.

2 - Senti falta de ver como é que as pessoas aplicam isso na pratica, mais exemplos pra poder me situar de como aproveitar ao máximo a ferramenta.

3 - Senti falta de outros exemplos porque aí eu poderia pensar nos exemplos e ver como aquele conhecimento que já foi aplicado poderia me ajudar a aplicar em uma nova solução;

Coding:

4C1 - Falta um suporte com exemplos para ver como os outros desenvolvedores aplicam na prática, pois poderia ajudar a se situar de como aproveitar o máximo da ferramenta (1,2,3).

D. O que você pensa sobre aplicações usando o G.A.M.E.?
Extraction:

1 - Com esses recursos que o framework traz, isso pode fazer o usuário a ter mais interesse pela ferramenta, colaborar mais, interagir mais, enfim, aumentar a participação.

2 - Acho que deixa o software mais divertido, e que as pessoas interajam mais, isso é positivo pra toda comunidade, não só pro usuario.

3 - A depender dos recursos, o software pode ficar meio carregado, o software tem que ser simples e introduzir estrategias que deixe o uso mais facil.

Coding:

4D1 - O software seria mais divertindo, atraindo o usuário pra ter interesse pela aplicação, colaborando e interagindo mais, aumentando assim a participação (1,2).

4D2 - O software pode ficar sobrecarregado de recursos (3);

E. Quais são os principais riscos associados ao uso do G.A.M.E.?
Extraction:

1 - Fazer a escolha certa de qual recurso vai introduzir no software para atrair o usuário.

2 - É relativamente simples, mas também é bem abranentep ra atender todo o processo de desenvolvimento, então pra mim que não sou engenheiro, acabo me confundindo um pouco.

3 - Dentre as técnicas de engenharia de software que conheço, conseguiu aplicar bem e fazer as escolhas certas.

4 - Você tem que conhecer esse processo para aplicar bem.

Coding:

4E1 - É preciso conhecer bem o processo para aplicar bem e fazer as escolhas certas pois apesar de ser simples, é bastante abrangente e pode confundir um pouco o desenvolvedor com menos conhecimento de engenharia de software (1,2,3,4).

F. Como você atenuaria esses riscos?
Extraction:

1 - [Sobre o risco de não saber como usar] Ter um recurso em anexo em que pudesse consultar casos de sucesso de gamification, algumas experiencias passadas, ter um histórico para guiar minhas escolhas de quais recursos de gamification vou usar.

2 - A questão da engenharia de software pode minimizar isso aí, talvez ter uma equipe recomendada pra usar gamification, ou uma pessoa com essas várias habilidades de engenharia de software de programação, game design ou um equipe com diferentes perfis.

Coding:

4F1 - Para atentuar o risco do conhecimento de engenharia de software, ter o recurso de exemplos para guiar as escolhas de quais recursos usar (1,2).

4F2 - Para aplicar bem o framework, ou uma pessoa que tenha habilidades de engenharia de software, programação e game design, ou uma equipe com esses diferentes perfis (2).

G. Você usaria o G.A.M.E. em sua vida? Por que?
Extraction:

1 - Sim, talvez pensar em como colocar gamification em uma rede social de robotica que eu uso.

2 - As tecnicas escolhidas, metodologias de engenharia de software, abriram um pouco a cabeça pra pensar sobre o desenvolvimento, devo voltar a olhar pra usar no meu cotidiano.

Coding:

4G1 - Sim, usaria em uma rede social de dominio especifico que participo (1).

4G2 - Desejo usar algumas das técnicas usadas no framework no meu cotidiano (2).

H. Quais etapas do G.A.M.E. se encaixam perfeitamente em seu propósito? Por que?
Extraction:

1 - A parte do gameplay que vocÊ coloca os quatro atores, quatro caracteristicas (player, action, state e feedback), acho que isso representa bem sua ferramenta

2 - A etapa de analise tambem dividida em cooperation, communication e coordination representa bem o fraemwork

Coding:

4H1 - O gameplay interaction model e dividir as atividades no 3C representam bem o framework(1,2)

I. Quais etapas do G.A.M.E. não se encaixam perfeitamente em seu propósito? Por que?
Extraction:

1 - Essa etapa de execução, impelmentação e resultado, talvez nao seja tao relevante pra metodologia em geral

Coding:

4I1 - A etapa de Execução não parece tão relevante pra metodologia em geral (1)

D.4 EXTRACTING AND CODING OF INTERVIEW WITH PARTICIPANT 5

A. O G.A.M.E. correspondeu as suas expectativas e porquê?
Extraction:

1 - Correspondeu parcialmente.

2 - Apesar de ser uma proposta interessante, está bem definidas as etapas, e estão claras as etapas e como elas se relacionam com as etapas anteriores e posteriores.

3 - A etapa onde você se baseia nos padrões de Game Design, acaba sendo muitas vezes muito extensa e as vezes até redundante.

4 - Talvez um subjconjunto daqueles padrões pudesse ser mais eficiente.

5 - Eu não achei que nenhum dos quatro testes de usabilidade se adequava a avaliação das alterações que eu tinha feito, porque eu imaginava que algum dos testes iria fazer uma comparação entre as duas soluções e ver se houve melhora em algum quesito.

6 - Não ficou muito clara a questão do state, ficou repetitivo, eu não consegui estebelecer muito bem um valor pra esse aspecto aí da modelagem.

Coding:

5A1 - Correspondeu parcialmente (1, 2).

5A2 - Tem etapas bem definidas (2);

5A3 - Etapas claras e como elas se relacionam com as etapas anteriores e posteriores (2);

5A4 - A etapa dos padrões de Game Design é extensa e redundante (3).

5A5 - Talvez um subconjunto daqueles padrões pudesse ser mais eficiente (3).

5A6 - Não achei que os testes de usabilidade se adequavam para avaliação das alterações (5)

5A7 - Não ficou clara a questão do state, ficou repetitivo, não consegui estabelecer muito bem um valor. (6)

B. O que você gostou ao usar o G.A.M.E.?
Extraction:

1 - Gostei de ser uma forma sistematizada, bem clada de aplicar gamificação em projetos digitais;

2 - Fornece todo processo, desde o inicio, da analise até o tesde usabilidade;

3 - É uma forma bem clara e completa, bem sistemática, sistematizada;

Coding:

5B1 - Gostou por ser uma abordagem sistematizada, bem clara e completa de aplicar gamificação em projetos digitais, fornecendo todo o processo desde a analise até o teste de usabilidade (1,2,3)

C. O que você não gostou ao usar o G.A.M.E.?
Extraction:

1. Tive dificuldade de entender desse aspecto de modelagem;

2. Não soube bem como escolher o teste, não achei que nenhum teste era adequado pra avaliar o que fiz;

3 - Executei o 3.2 antes do 3.1 pois eu achei que sabendo a forma de interação e qual é a ação e o feedback poderia escolher o padrão de game deisgn;

4 - Com relação aos padrões, achei que a lista é muito extensa e redundante;

5 - Talvez uma lista mais enxuta, juntar alguns padrões, poderia ajudar na hora de estabelecer os padrões;

Coding:

5C1 - Dificuldade em entender o state na modelagem (1);

5C2 - Não soube como escolher o teste, pois achou que nenhum era adequado (2);

5C3 - Escreveu as estórias antes de definir o padrão pois achou que sabendo qual é a ação e o feedback seria melhor para escolher o padrão (3);

5C4 - A lista de padrões é muito extensa e redundante, talvez uma lista mais enxunta e agrupada poderia ajudar nessa etapa (4,5);

D. O que você pensa sobre aplicações usando o G.A.M.E.?
Extraction:

1 - A aplicação envolva elementos de gamificação;

2 - Aplicação tenha uma boa usabilidade;

Coding:

5D1 - Espera que a aplicação envolva elementos de gamificação e tenha uma boa usabilidade;

E. Quais são os principais riscos associados ao uso do G.A.M.E.?
Extraction:

1. Uso condicionado pois quem trabalha com gamificação já está acostumado com certas formas de gamificar, então você acaba de uma certa forma sendo direcionado a usar essas mesmas formas;

2. Será que a lista de padrões está muito extensa ou eu desconsiderei padrões que pra mim não são usados?

3. Risco é de usar padrões de costume, reduzindo o beneficio do G.A.M.E. de diversificar;

4. Usar o teste de usabilidade errado pode resultar em resultados não reais, não verdadeiros;

Coding:

5E1 - Risco no uso condicionado de padrões baseado nas formas tradicionais de gamificar (1,3);

5E2 - Será que a lista de padrões é extensa ou desconsiderou os padrões não usados (2);

5E3 - O risco de usar um teste de usabilidade errado pode resultar em resultados não verdadeiros (4);

F. Como você atenuaria esses riscos?
Extraction:

1. Pra reduzir o risco dos testes, poderia fornecer pro usuário instruções de como escolher o teste;

2. Não sei como resolver o risco do uso condicionado;

Coding:

5F1 - Instruções de como escolher os testes poderia reduzir o risco do desenvolvedor escolher o teste errado (1);

5F2 - Não sabe como resolver o risco do uso condicionado (2);

G. Você usaria o G.A.M.E. em sua vida? Por que?
Extraction:

1. Sim, devido aos aspectos positivos;

2. Devido a estruturação da metodologia estar bem coesa, o processo de gamificação fica até mais rápido do que quando você começa a ter ideias de forma adhoc;

3. Além disso, ele direciona para os testes de usabilidade e analisar os resultados, não vai só até a modelagem, então você é capaz de avaliar aquele resultado, se a sua alteração funcionou, gerou algum resultado;

4. Não conheço muitas metodologias de gamificação, mas não vi nenhum tipo que fosse tão abrangente e ao mesmo tempo tão enxuta e prática;

5. Etapas bem coesas e bem relacionadas, por isso eu usaria na minha vida profissional sim.

Coding:

5G1 - Usaria na vida profissional devido a estrutura da metodologia estar bem coesa com etapas bem relacionadas (1,5);

5G2 - O processo de gamificação fica mais rápido do que de forma adhoc (2);

5G3 - Direciona para os testes de usabilidade e analisar os resultados (3);

5G4 - Não conhece uma metodologia de gamificação tão abrangente e ao mesmo tempo enxuta e prática (5);

H. Quais etapas do G.A.M.E. se encaixam perfeitamente em seu propósito? Por que?

Extraction:

1. Todas etapas com exceção das etapas 1.3 e 2.1.1 foram relevantes pro meu projeto.

2. Interessante a primeira etapa de objetivos individuais e de comunidade porque você precisa considerar esses aspectos na criação o projeto, melhoria ou criação de grupo;

3. Atividades cooperativas, problemas relacionados, vai precisar fazer o seu produto considerando o que o usuário vai querer mas também o que a comunidade precisa que os usuários forneçam;

4. Funcionalidades são importantes pra ter uma noção de quais recursos seu produto fornece e como você pode melhorar esses recursos.

5. Issues é uma das coisas mais importantes, pois você precisa direcionar seu projeto pra solucionar problemas;

6. A etapa 2.1.2 está muito relacionada com a etapa 1.2, então os recursos que a ferramenta oferece vão permitir que o usuário faça essas ações;

7. Você classificando os problemas pra atacarm elhor individualmente, então também peças importantes e estruturas;

8. Você modela a interação do gameplay, o que o usuário queria enquanto jogador e depois indificar os padrões que poderiam ser aplicados, é uma das etapas mais significativas;

9. Planejamento você escolhe o teste pra poder executar e avaliar seu resultado;

10. 4.1 E 4.2 são bem ligados, me confundiria até, também poderia ser uma etapa só.

Coding:

5H1 - Todas etapas foram relevantes, com exceção das etapas de Tecnologia e Crowdsourcing Target Problem (1);

5H2 - A primeira etapa de objetivos individuais e comunitários ajuda a considerar esses aspectos no projeto (2);

5H3 - Atividades colaborativas e os problemas relacionados são necessárias para considerar o que o usuário quer e a comunidade precisa (3);

5H4 - Funcionalidades são importantes para ter noção dos recursos do produto e como melhorar (4);

5H5 - Issues é uma das coisas mais importantes para direcionar o projeto para resolver os problemas (5);

5H6 - Etapa Crowdsourcing - What users do está bem relacionada com a etapa de Funcionalidades, pois os recursos vão permitir os usuários realizar essas ações (6);

5H7 - Classificar os problemas é melhor pra atacar individualmente (7);

5H8 - A modelagem da interação do gameplay, o que o usuário quer e identificar os padrões é uma das etapas mais significantes (8);

5H9 - Planejamento escolhe o teste pra executar e avaliar, etapas bem ligadas, poderiam até ser uma só (9,10) ;

I. Quais etapas do G.A.M.E. não se encaixam perfeitamente em seu propósito? Por que?

Extraction:

1 - As etapas 1.3 e 2.11 foram as menos significativas nam inha execução.

2 - A etapa 1.2 porque a tecnologia em nenhum momento influeciou nas etapas posteriores, não sei se entendi erradou ou se é comum, mas essa informação não foi usada em momento algum;

3 - A etapa 2.1.1 também não, porque ela ficou implicita, achei redundante, ela estava implicita nas funcionalidades e no que os usuarios fazem.

4 - Não utilizei a 2.1.1

5 - Todas as informações, com exceção da 1.3 e 2.11 eu utilizei novamente.

6 - Executei a etapa 4.3 antes da 4.1 e 4.2, defini como seria a solução antes de definir como seria a avaliação;

7 - Porque eu imaginei que o teste de usabilidade estaria muito relacionado com a mudança que você fez no produto;

8 - Me pareceu mais lógico fazer 3.2 antes do 3.1 porque a estória meio que diz o que o usuário espera, o que ele faz, o que ele é, qual ação que ele vai executar e qual feedback vai ter pra depois definir qual padrão iria usar pra contemplar aquela estória;

9 - Acho que se eu definir o padrão antes da estória, eu acho que a estória vai ficar condiciona ao padrão que foi definido;

10 - Acho que a estória vai restringir o conjunto de padrões que vai usar, mas acho que é menor;

11 - Eu acho que a etapa critica está aqui, que pra mim foi 3.2 e depois 3.1, quando eu deifni a estória e fui passar pro padrão, eu já fui meio direcionado, mas não porque eu defini a estória primeiro, e sim porque eu já estava habituado a outras soluções de gamificação;

12 - Se eu tivesse mais tempo pra trabalhar nessa solução , talvez eu tivesse utilizado outros padrões, talvez tivesse sido mais ousado em relação aos padrões utilizados.

Coding:

5I1 - Etapas de Tecnologia e Crowdsourcing Target Problem foram menos significativas pois ao contrário das demais, eu não usei em nenhum momento (1,2,3,4,5);

5I2 - Executou a etapa de Implementação antes das etapas de Planejamento e Instruções pois imaginou que o teste de usabilidade estaria ligado com a mudança no produto (6,7);

5I3 - Executou a etapa de Estórias antes da de padrões, pois sabendo qual ação e o feedback seria melhor definir o padrão pra contemplar a estória (8, 9,10);

5I4 - Talvez a etapa critica tenha sido a de passar pro padrão, pois devido a já estar condicionado a usar as mesmas soluções de gamificação (11);

5I5 - Se tivesse mais tempo pra trabalhar na solução talvez tivesse sido mais ousado e utilizado outros padrões (12);