Revista de Gestão e Projetos - GeP
e-ISSN: 2236-0972
DOI: 10.5585/gep.v8i2.528
Data de recebimento: 16/02/2017
Data de Aceite: 27/06/2017
Organização: Comitê Científico Interinstitucional
Editor Científico: Emerson Antonio Maccari
Avaliação: Double Blind Review pelo SEER/OJS
Revisão: Gramatical, normativa e de formatação

AGILE DESIGN: A COMBINED MODEL BASED ON DESIGN THINKING AND AGILE METHODOLOGIES FOR DIGITAL GAMES PROJECTS

ABSTRACT

Traditional approaches to project management seek to discipline the execution and to organize different activities to be performed in order to reach the project goals. However, that focus may not be suitable for the beginning of projects that require creativity, as in the case of digital games. This article aims to develop a project management model that covers the whole process of digital games development, by combining two approaches, Design Thinking and Agile Methodology. The proposed model was constructed based on the theoretical framework of both methodologies, and has three parts: the Design Thinking phase, the Connection phase and the SCRUM phase. In order to evaluate the pertinence of the model, a qualitative exploratory study was conducted through semi-structured interviews with game developers of the Brazilian market. It was observed that many aspects of Design Thinking are used in the game development, that agile development is fully diffused in this process, and that association of both methodologies, besides being possible, can contribute to the efficiency of this process. The main contribution of this work is the presentation of an integrated model of project management that fully attends the process of the digital games development.

Keywords*:* Game Development, Design Thinking, Agile Methodologies; Project Management.

PROJETO AGIL: UM MODELO COMBINADO COM BASE EM PENSAMENTOS DE DESENHO E METODOLOGIAS AGILAIS PARA PROJETOS DE JOGOS DIGITAL

RESUMO

As abordagens tradicionais de gestão de projetos procuram disciplinar a execução e organizar as diferentes atividades. Porém, esse foco pode não ser adequado para o início de projetos que exijam criatividade, como no caso dos jogos digitais. Este artigo tem como objetivo elaborar um modelo de gestão de projetos que cubra todo o processo de desenvolvimento de jogos digitais, através da combinação de duas abordagens, o *Design Thinking* e a Metodologia Ágil. O modelo proposto foi construído a partir do referencial teórico de ambas as metodologias, e possui três partes: fase do *Design Thinking*, fase de conexão das abordagens e fase do SCRUM. Para avaliar a pertinência do modelo foi realizado um estudo qualitativo de caráter exploratório, através de entrevistas semi-estruturadas com desenvolvedores de jogos do mercado brasileiro. Foi observado que muitos aspectos do *Design Thinking* são utilizados no desenvolvimento de jogos, que o desenvolvimento ágil está plenamente difundido nesse processo, e que associação de ambas as metodologias além de ser possível, pode contribuir para na eficiência de todo o processo. A principal contribuição do trabalho é apresentação de um modelo integrado de gestão de projetos que atende por completo o processo de desenvolvimento de jogos digitais.

Palavras-chave*:* Desenvolvimento de Jogos; *Design Thinking*; Metodologia Ágil; Gerenciamento de Projetos.

Marcelo Makoto Higuchi[1]
Davi Noboru Nakano[2]

[1] Bacharel em Engenharia Mecânica - Ênfase em Mecânica Plena pela Universidade de São Paulo - USP. Brasil. E-mail: **marcelo.m.higuchi@gmail.com**
[2] Doutor em Engenharia de Produção pela Universidade de São Paulo - USP. Professor da Universidade de São Paulo - USP. Brasil. E-mail: **dnnakano@usp.br**

1 INTRODUCTION

The digital game market was worth around US$ 100 billions in 2016 (Research, 2016), and game revenues have surpassed well established entertainment businesses as the movie industry. In Brazil, the digital game market in 2015 had approximately 33.6 million users, about one sixth of Brazilian population, placing Brazil as the 11th global market.

The development of digital games has two aspects: one is associated to software development, i.e. programming and coding, while the other is related to a more artistic perspective, e.g., graphic and character design, music, etc. Thus, many researchers as Akta & Orcun (2014), Albino, Souza, & Prado (2014), Hodgson & Briand (2013), Manninen & Kujanpa (2006) argue that game development can be considered a especial case in software development. Akta & Orcun (2014) consider that game development differs from software development essentially because of the creative aspect, that involves the game concept, aesthetics and entertainment.

Game creation is, by nature, an unstructured process. Historically, this process has been conducted following an intuitive and organic path (Brathwaite & Schreiber ,2009; McCarthy & Byron, 2005). However, recently, some researchers, as Albino, Souza & Prado (2014), observed that the development process could be managed following project management practices, which involves task planning, scheduling and coordination, taking into account deadlines, development costs and quality standards.

However, the literature on game development methods has focused mainly in the technical aspects of game development, the management practices usually performed in the intermediate and final stages. The initial stage, when creative content is developed, still need specific studies.

There are approaches that deal with managing complex, ill-defined projects that also require creativity. Many of them focus on attending user's needs, like Design Thinking (DT), an approach applied to solve complex problems, as in game development, in which there is often no clear definition of how best meet player's needs.

This article proposes a model for digital game development that combines DT with agile methods for project management. Thus, the model can be applied to both the creative and technical stages of game development, covering all the stages of the game development process. Also, as the digital game market became very dynamic and increasingly competitive, game developers, to a certain extent, have paid more attention to the value proposition of their titles and less on the game development process, which reinforces the usefulness of the model suggested in this article.

The remaining text is structured in four sections: a literature review on Design Thinking and Agile Methodologies (AM), game development and software design, and the proposed model; a description of the methodology of this study; the analysis and discussion of results; and a conclusion section, with ending remarks, limitations and possibilities for future research.

2 LITERATURE REVIEW

2.1 Design Thinking Approach

Although the term Design Thinking was coined in the 1980s by Aldridge (1980), it only gained relevance in the late 2000s (Fleury, Stabile, & Carvalho, 2016), with Brown´s (2008) research. He argues that DT can be considered a discipline which encourages practitioners to apply a user centric approach, as designers do, and to apply their methods for solving complex or poorly structured problems, known as Wicked Problems (Rittel & Webber, 1973) or Ill-Structured Problems (Simon, 1973).

Ill-Structured and Wicked Problems

A Wicked Problem (WP) cannot be well defined, separate into parts, and accept an infinite number of solutions. It is characterized by confusing and conflicting information, which makes them difficult to be structured (Rittel & Webber, 1973).

A Ill-structured problem (ISP) presents a poorly defined structure (Simon, 1973). Thus, in order to characterize it, one has to define what a Well-Structured Problem (WSP) is in that context, to then oppose it to an ISP: all problems that do not have all the characteristics of a WSP can be understood as ISP. Solutions for ISPs can be proposed by experts, e.g. designers, who are able to make logical connections and to find possible best alternatives among the infinite possible solutions.

Kimbell (2011) argue that both WPs and ISPs are directly associated to Design Thinking and Buchanan (1992) defined a typical DT problem: the creation and development of a non-existing product or service.

According to those definitions, one can affirm that the conception of digital games is both an ISP and a WP, as it has infinite possible solutions, there is no obvious and well-defined criterion for quality and performance evaluation and it contains confusing and conflicting information, which hinders its structuring.

Different approaches of Design Thinking

DT has been associated with resolution of ISPs and WPs (Buchanan, 2014, 2016; C. H. Dorst, 2006; K. Dorst, 2011). There are several definitions for DT, and some authors proposed a comprehensive definition, using a semantic analysis of the existing literature (Fleury, Stabile, & Carvalho, 2016):

> "Design Thinking is a human-centered approach applied to wicked problem-solving that starts with the understanding of different users' perspectives. It involves multidisciplinary teamwork based on the balance between cooperation-conflict among different actors in a co-creation process, in which the conflict of ideas become the genesis for the establishment of innovative solutions."

However, as there is more than one definition for DT, there is also more than one way to describe DT's approach (Kimbell, 2011, 2012): (a) Design Thinking as a cognitive style, (b) as a general theory of design, and (c) as an organizational resource. Each of these descriptions has a specific focus: in the cognitive style approach, the focus is placed on designers and experts (Cross, 1982, 2006; C. H. Dorst, 2006); in the general theory of design the subject is discussed as discipline or an autonomous field (Buchanan, 1992); and in the organizational resource approach, DT can generate innovation for businesses (Brown, 2009; Martin, 2009). In this text, DT is understood following the last view, as a resource for organizations, since the digital game market is very dynamic, and thus, innovation is essential for development studios.

DT approach as a resource for organizations

As consequence of increasing market competition, the use of more effective design approaches can be considered as essential for any company. In this line, many authors emphasize the use of designers' skills to business problems such as new product development and the re-assessment of production processes as effective solutions (Dorst, 2011; Kimbell, 2011; Martin, 2009; Razzouk & Shute, 2012). Other authors understand DT as a human-centered methodology that facilitates innovation (Bauer & Eagen, 2008; Brown, 2009; Dunne & Martin, 2006), essential for competitive advantage, and Martin (2009) even states that the use of DT is mandatory for companies seeking innovations, since it balances analytical skill to intuition.

Thus, the use of DT has received attention in the management community. They feel the immediate need to differentiate their strategies to solve complex and ill-formulated problems experienced by their organizations (Brown & Katz, 2011; Stacey, Griffin, & Shawn, 2002). For example, Lockwood (2009) illustrates that by applying DT it is possible to discover unmet consumers' needs, and to generate competitive edge for the company.

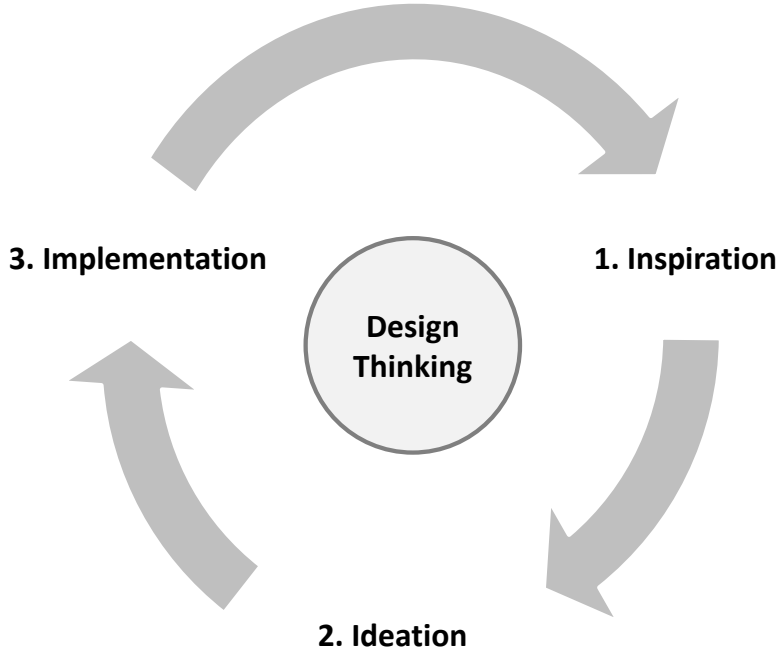
Designers or practitioners that use a DT approach should have the following characteristics (Brown, 2008):

- Empathy: being able to visualize the world from diverse points of view, such as colleagues, customers, etc.;
- Integrative thinking: using a bird´s eye view, considering all relevant points for problem-solving, even when they are contradictory;
- Optimism: no matter how complex problems are, there is always a potential solution that is better than existing ones;
- Experimentation: proposing questions and exploring its constraints in a creative manner, which that can lead to entirely new directions;
- Collaboration: working collaboratively and having multidisciplinary knowledge.

Stages of the Design Thinking

According to Brown (2008), every *design* process must go through three stages: Inspiration, Ideation and Implementation (Figure 1).

Figure 1 - Design Thinking stages



Source: elaborated by the authors, based on (Brown, 2008)

In Inspiration the project questions are formulated, which are the motivations for searching for solutions. These questions always focus on the fulfillment of end-user requirements in relation to the product to be elaborated. For example, Bonini & Sbragia (2011) consider that all possible questions can be summarized in a single question, which would be: "What is the solution that my clients need? ".

Ideation is characterized by the generation, development and testing of ideas, looking for answers to the issues raised during the Inspiration phase. In this step best ideas are generated and chosen in order to accelerate the prototypes creation. The test of ideas and solutions is done through simple prototypes. According to Brown (2009), the focus of this stage is the creation and conceptualization of the solution and, therefore, it is the more fundamental step to formulate and improve a solution. And, as pointed by Bonini & Sbragia (2011), this process alters traditional thinking, which first imagines a solution, designs it, to then build a product or a service. In DT an idea is tested quickly, and the design is essentially based on experimentation.

After defining the solution to be pursued, in the last phase, Implementation, the concept is developed and the final product is created, which is placed in market.

This process is the most usual in the literature. However, there are other DT approaches, such as the *Bootcamp Bootleg* of the Stanford University d.School (2008), which considers the existence of five stages: Empathy, Definition, Ideation, Prototyping and Testing. Another process is the HCD (Human Centered Design), which contains seven steps: Observations, Narratives, Themes, Opportunities, Solutions, Prototypes and Implementation Plan (IDEO, 2011).

According to Larsen e Majgaard (2016) game design has received considerably attention in the last years. However, there is little research that explored the use of Design Thinking in the game development process, as it's shown by Hayes & Games (2008) in their literature review. The main result of Hayes & Games (2008) is that existing software and educational applications generally emphasize the production phases, like programming, over the creative ones, like creating concepts associated to game design. Thus, the authors stressed the potential value of the use of DT in game development education. These findings reinforce the usefulness of the model developed in this text, which

will cover both the creative and the production phases of the game development process.

Brown (2009) illustrated other example that shows the relevance of the Design Thinking in the video game industry, studying the Nintendo Wii case. Before Wii, most of video game developers focused mainly on hardware development. However, Nintendo developers looked first for clients' needs, thinking how they could create a video game more appealing to a wider market. This illustrates how a different design approach, design thinking, has had significant relevance in Nintendo Wii success.

Through the literature analysis it was shown that the DT process focuses on construction of creative solutions focused in the clients' needs, leaving, however, management and production tasks to a second plane. Thus, an integrated model as proposed here is needed to attend the entire game development process.

a. Agile Methodologies

The Agile Manifesto, elaborated in 2001, revolutionized software engineering (Dingsoyr, Nerur, Balijepally, & Moe, 2012). The creators of this manifesto, a group of practitioners and academics, discussed how to make software development faster and more efficient. As result the Agile Manifesto was elaborated, containing four fundamental principles (Dingsoyr, Nerur, Balijepally, & Moe, 2012; Agile Alliance, 2001):

- Collaborative development, with privileges and process restrictions agreement;
- Lean mentality to reduce unnecessary work, e.g., excessive documentation;
- Clients and employees are key to improve the quality of projects;
- Uncertainties are always present in software development, and the attempt to control variations using statistical analysis has limited results.

Main focus and characteristics

Many agile methods for project development were proposed: Extreme Programming (XP), SCRUM, Crystal Methods, Dynamic System Development Method (DSDM) and Feature-Driven Development (FDD). Research has also studied several aspects such as: the adoption of agile methods in projects (Boehm, 2002); the effectiveness of pair programming in software development (Williams, Kessler, Cunningham, & W., Jeffries, 2000); trust, self-organization and communication in development teams (Moe & Dingsoyr, 2009) ; test-driven development results (Erdogmus, Morisio, & Torchiano, 2005); adoption and post-adoption issues of software development (Cao, Xu & Ramesh, 2009)

The main focus of agile methods is on fast delivery of high-quality software, and thus, iterative development, quality of products, etc., are considered as secondary aspects (Dingsoyr, Nerur, Balijepally, & Moe, 2012). Despite the single focus, many authors discussed agile development from different perspectives. For example, some authors understand that agility embeds the ability to quickly and flexibly respond to changes in the business environment and technology (Henderson-Sellers & Serour, 2005; Higsmith, Consortium, & Cockburn, 2001); it promotes maneuverability and fast response to client demands (Cockburn, 2007); it allows efficient incorporation of user requirement changes during the project (Lee & Xia, (2010); it allows fast change and learning, and high costumer value (Conboy, 2009).

Agile development methods are also recognized as flexible alternatives to traditional methods such as ISO 9000 and CMMI (Dingsoyr, Nerur, Balijepally, & Moe, 2012). Due to the contrasting characteristics between traditional and agile methodologies, many authors compared those approaches and also the combination of them (Dingsoyr, Nerur, Balijepally, & Moe, 2012). According to Albino, Souza, & Prado (2014), the agile methodologies of project management are different both qualitatively and quantitatively from traditional methodologies such as the PMBOK® ((PMI), 2008).

In the next subtopic is presented the SCRUM, an AM used in the model proposed in this work.

SCRUM

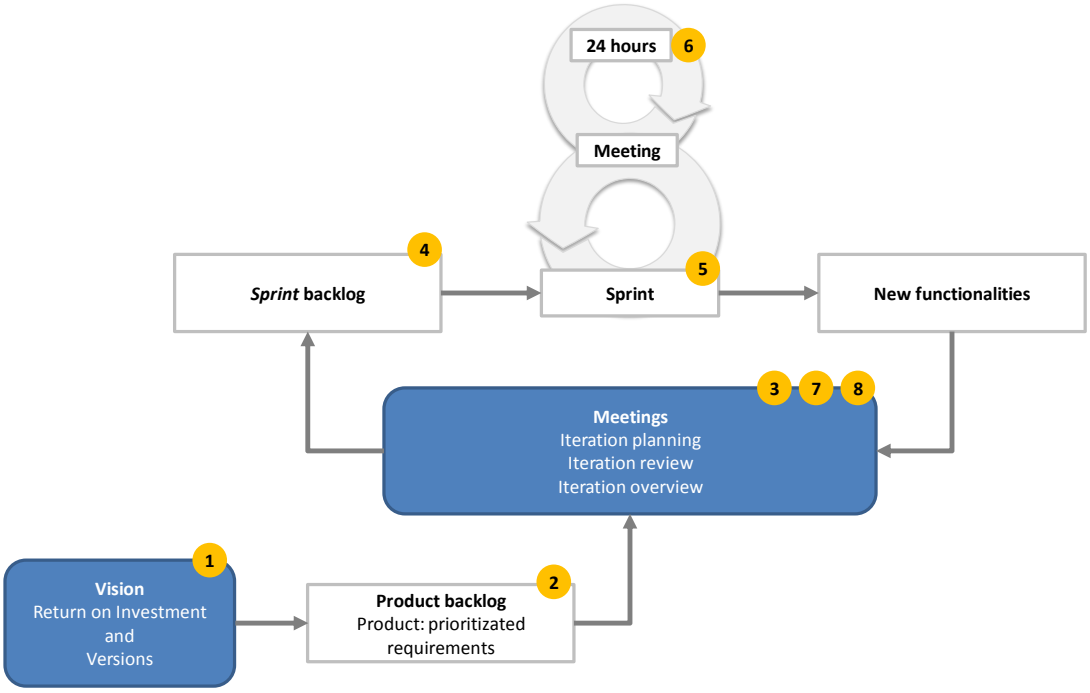
SCRUM can be considered one of the main, if not the main, agile methodology (Albino, Souza, & Prado, 2014). It was developed by Ken Schwaber and Jeff Sutherland in the early 1990s. It defines a process in which projects advance in iterations, known as *sprints*. There are three key actors in SCRUM (Schwaber, 2004):

- Product Owner: represents the project demand, in some cases the client, he typically controls project requirements and defines their priority;
- SCRUM Master: is the project manager, he applies SCRUM to different projects, and he uses his understanding on project goals to assist the team;
- SCRUM Team: the team is responsible for the project development, they help to determine project effort, they can self-organize, separating the requirements

requested by the Product Owner into tasks and deciding who is responsible for each of these tasks.

According to Cooper (2014) the SCRUM methodology can be understood as an incremental process that allows to demonstrate the product to the client by performing work packages (*sprints*). Figure 2 and Figure 3, explain SCRUM in more detail, highlighting its iterative cycles and the main actors of each step until product development conclusion.

Figure 2 - SCRUM stages



Source: elaborated by the authors, based on (Schwaber, 2004)

Figure 3 - SCRUM stages

#	STAGE	STAGE DESCRIPTION	MAIN ACTORS
1	Vision	Product Owner's view about the product to be developed. It can start as vague vision, both in market and technical aspects, but it will become clearer with the project progresses. The PO is responsible for assuring vision and maximizing project funders´ Return On Investment (ROI).	Product Owner (PO)
2	Product backlog	Items list, of functional and non-functional requirements, and corrections. During the project, items are added, removed, and their priority may change.	Product Owner
3	Iteration (*Sprint*) planning meeting	Meeting in which the PO, SM and ST decide what will be accomplished in the *Sprint*. In the first stage the PO presents what he wants in the "Product backlog" and the ST defines what can become a functionality in the next iteration, generating the "Sprint backlog". In the second part of the meeting the ST defines the tasks required to implement each item of the "Sprint backlog".	Product Owner, SCRUM Master (SM) and SCRUM Team (ST)
4	*Sprint backlog*	Contains selected items from the "Product backlog" and the tasks required to turn those items into functionalities. During the *Sprint* only the SCRUM Team and Master SCRUM can change this list.	SCRUM Master and SCRUM Team
5	*Sprint*	A 30-day consecutive period in which ST and SM work to produce what was defined at the "Iteration planning meeting"	SCRUM Master and SCRUM Team
6	SCRUM daily meeting	15-minute meetings are held to check the progress and difficulties faced by the ST. The SM conducts the "SCRUM daily meeting" and ST's members should answer three questions: What have you done since the last daily meeting?; What will you do until the next daily meeting?; What were the problems faced in the accomplishment of your work?	SCRUM Master and SCRUM Team
7	Iteration review meeting	Meeting conducted by the SM and held at the end of the iteration, which ST presents to the PO what was accomplished.	Product Owner, SCRUM Master and SCRUM Team
8	Iteration overview meeting	Meeting held by SM, ST and PO, aiming to raise what was good in the last iteration and what could be better in the next.	Product Owner, SCRUM Master and SCRUM Team

From the Figure 2, one can affirm that the development process is iterative, thus, the vision and

work list can be altered at any moment during the project (Cao, Xu & Ramesh, 2009).

The steps shown in Figure 3 can be grouped in four phases, as follows:

- Planning: steps 1, 2 and 3 are associated with overall project planning, from the requirements and values to the planning of the *sprints*;
- Construction: steps 4, 5 and 6 are when tasks are defined for the construction of product characteristics during *sprints* and also daily meetings are conducted to assess difficulties and create empathy among team members;
- Inspection: step 7 is when product characteristic developed in the *sprint* is presented to the final customer for evaluation; and
- Adaptation: step 8 is when improvement points are defined and necessary adjustments and adaptations will be executed in later sprints.

At the end of all *sprints*, the Product Owner will receive a high quality final product, to which he actively contributed for its development.

b. Association of game development with software engineering

As discussed by Albino, Souza, & Prado (2014) the development of games, or *game design*, can also be considered as a process of software development, since both processes have many similar stages.

According to Sloper (2002) the process of creating digital games has five stages: design, pre-production, production, post-production and post-release. The software development process has the following activities (Sommerville, 2003): software specification, software design and development, software validation and software evolution.

It is possible to list the similarities between the two processes, such as the "pre-production" and "software specification" steps that are made the requirements and final product definitions, for example, the setting the game platform, graphic properties, programming language etc.; "production" and "software design and development", both stages are associated with the process of development of the final product; "post-production" and "software validation", stages associated with final product testing and refinements; "post-launch" and "software evolution", steps considering the use and consequent product evolution with the intention of keeping it continually attractive to its end customers.

However, the first process in game design "conception" is not contemplated in software development process. It can be carried out either individually or in a group: when performed collectively, it is usual to use the brainstorm technique, in which various ideas such as: themes, game play, target audience, platform etc. are discussed (Berthêm, 2007). This stage requires creativity, and is not addressed in software development methods.

c. Proposal of integrated model: Design Thinking and Agile Methodologies

The game development process has both artistic, creative, and technological characteristics, which allow to integrate the two approaches: Design Thinking and Agile Methodologies. Both offer competitive advantages, in product differentiation and in cost efficiency. The Design Thinking approach can be applied to the creative aspect of the game development process, while Agile Methodologies can be used on the prototyping and development of the final product.

In addition, the literature shows that both approaches have many similarities, which strengthen their combined use in digital games development. Figure 4 highlights similarities between DT and AM.

Figure 4 - Synergies between *Design Thinking* and Agile Methodologies

CHARACTERISTIC	DESIGN THINKING	AGILE METHODOLOGY
Solution of poorly structured problems	Essentially associated for solving poorly formulated and complex problems (Buchanan, 1992; Kimbell, 2011; Dorst, 2011; Razzouk & Shute, 2012).	As uncertainties are part of software development, thus project developers have to deal with poorly structured problems (Dingsoyr et al. 2012; Lalsing 2012; Nerur, Mahapatra, e Mangalaraj 2005) .
Customers desires are key to project development	All DT approaches are essentially human-centered and emphasize user experience (Bauer & Eagen, 2008; Brown, 2009; Dunne & Martin, 2006).	Final customers and project collaborators are active in final product development, thus ensuring a better quality and suitability of the final product (Abrahamsson, Conboy, & Wang, 2009; Siau, 2005; Williams & Cockburn, 2003).
Iterative productive process	The DT process is iterative, there is no clear barrier between stages, iteration are fundamental for solution development (Brown, 2008; Fleury, Stabile, e Carvalho, 2016).	Agile processes are mainly iterative, e.g., SCRUM Sprints (Williams & Cockburn, 2003; Schwaber, 2004).
Team collaboration (interdisciplinary and multidisciplinary)	DT practitioners work in teams, being characterized by their interdisciplinarity and multidisciplinarity (Johansson-Skoldberg et al., 2013; Seidel e Fixson, 2013).	Developers who use AM work fundamentally in multidisciplinary teams, composed by programmers, project managers, testers etc. (Dingsoyr et al., 2012; Moe & Dingsoyr, 2009).
Fast prototyping	Fast prototyping is fundamental in the Ideation stage, the use of simple prototypes allows to identify new creative and innovative ideas. (Liem & Brangier, 2012; Vetterli et al., 2013).	AMs focus on the most efficient development of projects, assuring a high quality product and cost efficiency (Cockburn & Highsmith, 2001; Henderson-Sellers & Serour, 2005).

Considering the game design process presented by Sloper (2002), the combination of DT with AMs proposed in the present work (Figure 3) considers that the initial stages "Conception" and "Pre-Production" follow a DT approach, whereas the final stages, "Production", "Post-Production" and "Post-Launch", are conducted using AMs.

Figure 3 - Proposed model for game development combining *Design Thinking* and Agile Methodologies

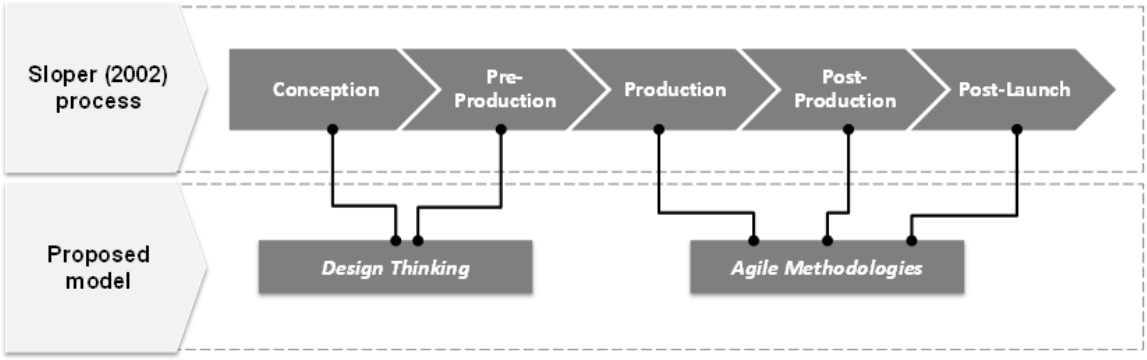


Figure 3 represents the first insight of the proposed model that will be described with more details in the next section. The model was created considering that game development is typically unstructured; thus, it's suggested a combined model to improve the game development through a project management using DT and AM.

Specifically, DT is associated with the Conception and Pre-Production stages of Slopers (2002) process, to help game developers to work in their ideas in a deeply and structured form. The others steps of Slopers (2002) process use, the most used AM methodology in the game development process.

3 METHODOLOGY: ASSESSMENT OF DT AND AM USE BY DEVELOPERS

To evaluate the degree of knowledge and use of Design Thinking and Agile Methodologies by developers, an exploratory, qualitative research approach was adopted, to better understand the process of game development. Semi-structured interviews were conducted with five Brazilian game developers, working in different companies. For the selection of respondents, two criteria were considered: (a) experience in game developing and (b) a leading role in the company, e.g., as a manager or partner.

Company A has launched its first game recently, for cell phones, and its game already have more than a 100 thousand in just three months. Company B can be considered as the most successful in the sample, obtaining more than 100 million downloads of one game, but now its founders are focused on another market, thus it has not released more games in recent years.

Although the Company C was created very recently, the respondent is an experienced developer, with more than 6 years in the business. Company D develops games for cell phones and computers and, recently, its developers are concluding its most ambitious game. Company E recently changed the target platform, it used to develop products for cell phones and computers, and now it's developing virtual reality applications. Figure 4 provides a consolidated description of respondent firms.

Figure 4 - Companies' characteristics

	COMPANY A	COMPANY B	COMPANY C	COMPANY D	COMPANY E
Comapany's age	2,5 years	10 years	3 months	4,5 years	1,75 years
Games launched	1 game	More than 50 games	7 games	9 games	6 games
Developers background	Systems information, game designers etc.	Engineers, business administrators, designers etc.	Computer scientists and industrial designers	Game designers	Engineers, game designers etc.
Platform target	Cellphones	Cellphones	Computers and consoles	Cellphones and computers	Virtual reality devices

Questions were structured in two large blocks: socioeconomic information and project management questions about DT and AMs. In the case of the DT, the questions had two purposes: to verify if the game developers were familiar with it, and to verify if they already had used techniques related to that approach. Technique selection was based on the literature, and can be found in Figure 5**Erro! Fonte de referência não encontrada.**. All interviews were in person and hand notes were taken.

Figure 5 - Stage and techniques of Design Thinking application

STAGE	RELATED TECHNIQUES	DESCRIPTION
Inspiration	Strategic challenge	Proposing questions such as "How can we..."
	Challenge selection	Evaluation and selection of challenges to be pursued by the team
	Knowledge sharing	what is and what is not known for solving the problem, and what needs to be studied to solve the proposed question
	Research planning	Research planning considers: users, clients, experts, contexts and benchmarks
	Questionaires	Development and application of questionnaires to users, clients and experts
	Research	Research of benchmarks and problem contexts
Ideation	Sharing	Sharing the information and perceptions found in the "Inspiration" phase, and possible solutions
	Personas	Creation of fictitious characters, who serve to refine the possible solutions
	Empathy map	Understanding personas, what they feel, see, hear, speak and do, their weaknesses and their strengths
	Synthesis	Synthesizing all learning

	Brainstorming	Creation of possible solutions
	Customer Journey	Description of the user's journey taking into account his thoughts and feelings; allowing to propose possible solutions
	Blueprint	Visual notation that represents the solution process, including actors and activities
Implementation	Prototyping	Creation of a prototype that expresses all practitioners' ideas
	Hyphoteses and tests	Hyphoteses for solution, and tests for validation of hyphoteses
	Pivoting	Solution change based on the results of performed tests
	Considerations	Main considerations of the group of practitioners on the solutions' development

Source: Elaborated by the author from (IDEO, 2011)

Regarding the questions associated to AMs, which game developers are more familiar to, they focused on the frequency and phase of the development process in which they are used. In this case it was asked directly if they knew and used an AM, and at which of Slopers (2002) process stages they started to be used.

4 ANALYSIS AND RESULTS DISCUSSION

General vision about the Design Thinking approach

All interviewees were somehow familiar to Design Thinking. One interviewee watched a short internet video about DT while another participated in a course on the subject. However, even after taking the course, he was not able to apply it for game development. Remaining respondents had poor knowledge on DT.

However, from the interviews and the literature review, it was observed that the game development fits all characteristics listed by Brown (2008) for DT:

- Empathy: game developers think like players, always seeking the highest satisfaction of their desire;
- Integrative thinking: game developers think in an integrated way, associating game features to client demands, e.g., market segment, how they will meet the player's needs, which kind of control mechanisms they will use, how to maintain and expand their customer base etc.;
- Optimism: developers are always optimistic about their solutions, as they strive to meet their own desires, which, to a certain extent, are similar to users' demand;
- Experimentation: in the game development process it's common that many prototypes are created and tested;
- Collaboration: game development is by nature collaborative and multidisciplinary; hardly ever a game is fully developed by a single person and it requires experts from various areas, such as animation, creation etc. A typical developer profile is one that is holds expertise in one area and superficial knowledge on about other areas, which facilitates collaborative work.

To evaluate the use of the techniques presented in Figure 5, it was created a classification to assess to what extent game developers use the technique. Four grades were created: (1) Complete, when the game developer use the technique totally, as described in Figure 5; (2) Partial, when the technique is partially used; (3) Residual, only some actions are employed in game development; and (4) None, when techniques are not used at all.

The result of the evaluation is presented in Table 1, which presents the percentage of all techniques and how much they are used. For example, in Company A, 44% percent of all techniques presented in Figure 5 are fully used. It can be seen that most of the techniques presented in

Figure 5 are used by the interviewees (all the interviewees use the techniques in a Complete and Partial form in approximately more than 69%).

Despite the positive results about the use of the DT techniques, it should be noted that the developers were unaware of their nomenclatures and also didn't know how to define them in a clear way. Also, they apply those techniques only partially in most cases. For example, in the "Research Planning" technique, all respondents stated that they use to plan, but they do it in a less formalized way.

In conclusion, DT techniques have been used by developers mainly associated with creative aspects of game development, and as expected, they are used them mostly in the Conception and Pre-production stages as it was indicated by the interviewees. As an example, in the creation of a new game, developers initially seek game benchmarks, informally, without documentation all the process.

Table 1 - Classification of the use of DT techniques

	COMPANY A	COMPANY B	COMPANY C	COMPANY D	COMPANY E
Complete	44%	44%	50%	31%	44%
Partial	44%	31%	25%	50%	25%
Residual	13%	13%	13%	13%	13%
None	0%	13%	13%	6%	19%

Use of DT's techniques

Table 1 illustrates that most of the DT techniques are used by developers, but some are not used at all. Those techniques are now listed, with the main reasons why they are not applied.

- Questionnaires: none of the interviewees use questionnaires to identify tastes and interests of their possible clients, because, in the majority of cases, they see themselves as their own clients, thus they use their opinion to develop their games;
- Personas: developers view themselves as their own personas, thus, they do not create a fictional person for game improvement;
- Empathy map: as personas are not created, this technique is also not used. Developers imagine themselves as their players, thereby, they can self-assess the weaknesses and strengths that will be used in the creation of their games;
- Customer Journey: the initial game scope is already well defined: the technical platform, game mechanics and type of users; therefore, developers already know, a priori, when and which platform players will use;
- Service Blueprint: none of the developers use blueprints, which involves both the user cycle and actors involved.

Although they do not specifically use such techniques, developers do take into account many aspects of them. For example, in the case of personas and the empathy map, as they consider themselves as possible clients, they already know how to define what would be a product that would fully meet their desires.

General vision about the use of Agile Methodologies

As expected all developers use agile methodologies in the development of their games. According to them, their use starts between the Pre-Production and Production stages of Sloper (2002) process. In addition, the most commonly used agile methodology is the SCRUM.

In the Pre-production stage the requirements of the games are defined in the same way as the in a regular software. For games development, specific requisites are: game mechanics, technical platform, game genre, target audience, graphic structure, game options etc. That step fits as the initial part of the software development, in the case not necessarily entering in the cycles of development of functionalities, such as SCRUM *sprints*.

The Production stage is conducted following development cycles, in which developers produce game sections and test them at the end of each cycle, evaluating its functionalities.

The use of SCRUM is fully diffused among the interviewed developers, who understand it as an essential part for their game development process, from the first prototype to the final code.

Approaches combination DT and AM

As the interviews showed that both techniques similar to DT and AM concepts are

commonly used by game developers, there is evidence of the possibility of integrating both approaches. Considering the combination of those two approaches there is great chance of improving game project management, especially in the initial stages, the creative ones, that nowadays do not have a structured process to explore and develop the game concepts ideas.

In fact, DT can be associated with the first stages of game production, which are characterized by high levels of creativity; while the final stages are characterized by the intensive development or implementation of ideas, when AMs can be used.

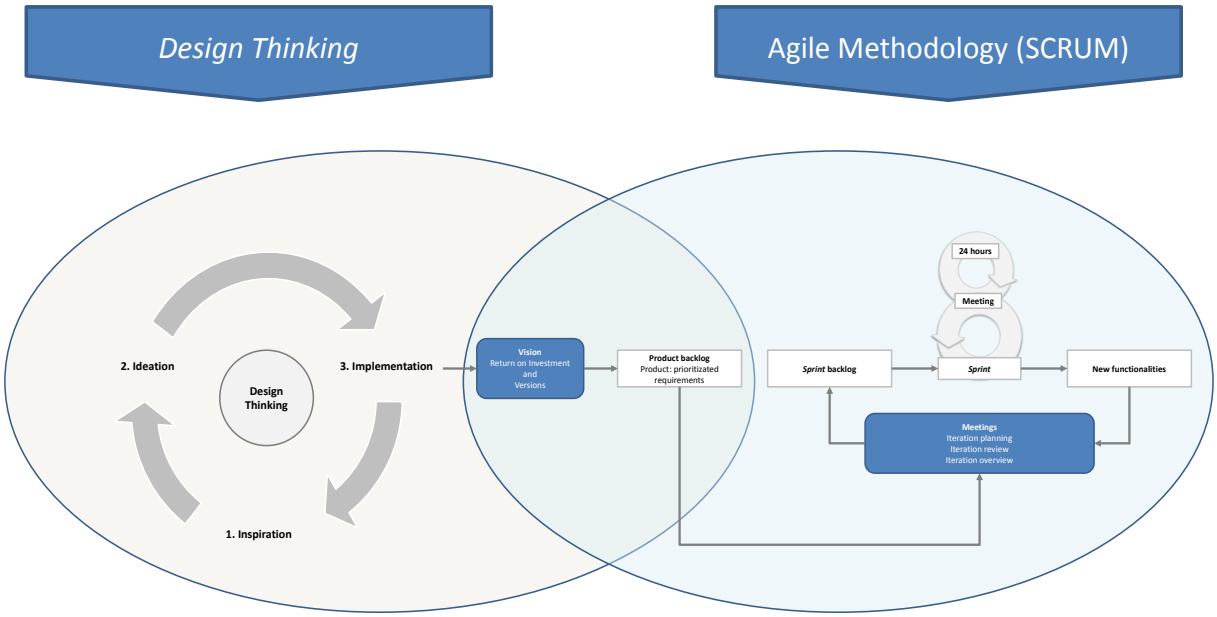
Thus, the main contribution of the proposed model, presented in the next section, is the addition of some DT techniques to help developers to structure and explore their ideas in a complete form. Besides, the game developers will have a project management approach to go through all the steps of game development process presented in Figure 6.

Proposed Model

Considering what was exposed, a model that combines DT and an AM methodology (in this case, SCRUM) is proposed, as depicted in Figure 2.

.

Figure 6 - Proposed methodology - *Design Thinking* & SCRUM



In Figure 6 is possible to see the proposed model of project management for game development, in a macro vision it has two phases, one associated to DT (left side) and other to SCRUM (right side). In the middle, the connection of these two approaches, that is, the end of DT process and the beginning of SCRUM methodology. In the following paragraphs the proposed model is described.

- **DT phase**
  - Inspiration: It's the beginning of the game development process, where developers willing to produce a new game, choose initial ideas about game type and share knowledge on similar games, game mechanics etc. After that, they define, through a plan, how to better understand the player's needs, using interviews, questionnaires and benchmarks;
  - Ideation: Phase when ideas will be improved or discarded, and also when game concept is developed. In this stage, developers will share all knowledge obtained in the Inspiration phase, and will brainstorm to define and synthesize better ideas for game concept. Thus, despite game developers do not use most of the techniques presented in Figure 5, they use is emphasized to assure empathy to game players;
  - Implementation: Last phase associated to DT before entering the SCRUM stage. This phase will help

game developers to refine their ideas through experimentation. During Implementation prototypes that depicts all hypotheses on game mechanics, scenarios, gameplay are created. Then, those prototypes are tested to validate or refute the game hypotheses, to find the best game concept.

- **Connection phase (DT + SCRUM)**
  - Vision and Product backlog: This is when SCRUM starts. In this phase, the development team will act as the client, who has a vision of the game concept and will define the product backlog. By the end of the DT process, game developers will have a clear vision of the game concept, ensuring a better definition of the product backlog;

- **SCRUM phase**
  - Iteration meeting: It is the point where decision on which parts of the game will be developed in the Sprint, considering production restrictions and team expertise;
  - Sprint backlog: it is the result of the "Iteration meeting", a formal list that contains selected items from the "Product backlog" and tasks required to turn those items into game parts;
  - Sprint: a 30-day consecutive period where game developers works to develop all items contained in the "Sprint backlog";
  - SCRUM daily meeting: short meeting in order to check the general progress and difficulties faced by the game development team;
  - Iteration review and overview meeting: meeting conducted by the game development manager (or SCRUM Master) held at the end of the iteration, where is presented the game part developed. Besides, it's discussed what was good in the last iteration and what could be better in the next.

In sum, DT techniques are used to improve the stages of the vision and work list, which are not managed with the use of SCRUM, as they are the most creative part of the game development process and reflect project owner demands. Vision stage is improved by using all techniques presented in Figure 5 as they give a broader view of the game concept, taking into consideration external opinions and not only the developer's view. The work list will be more assertive, as the creation of prototypes and hypotheses validation allow developers to advance towards a better specified and detailed game, ensuring both faster development and greater success of the game in question.

It's emphasized that the interviews were fundamental to validate the proposed model. First, it was seen that the game developers do not have any structured project management approach to develop the concept of the game and commonly they use similar techniques from DT, exposing that they have some familiarity with DT; considering what was pose the DT will help them in the project management of the creative part of the game development. Second, as expect the use of AM is diffused between the interviewees, mainly associated to the production parts of the game development process. Thus, the combined use of those two approaches in all the game development process could be considered as possible both in practice and theory.

5 CONCLUSION

This paper discussed game development process, an issue that currently needs more discussion. Game development should be essentially user-centered, as a game are should fulfill players' demands. Thus, the use of Design Thinking can contribute to the improvement of game design and production, because its techniques incite and explore new ideas. For example, the DT technique of developing personas create a view of the extremes of the user population, for instance, children and the elderly are the extremes of an age normal curve, and thus, if the developer can create a solution that meet demands from these two extremes it certainly will attend a larger number of players. As expected, game developers already use Agile Methodologies in their game development process. It was also observed that the most popular AM among developers is the SCRUM.

Through the interviews it was observed that, in fact, the DT was more associated with the initial stages of game development and Agile Methodologies to final stages, and also considering the synergies observed in the literature (Figure 4), it was proposed and validated a integrative model with both approaches DT and AM.

The model is depicted in Figure 4: the game development process begins with focus on DT practices to then use AMs practices, for instance, SCRUM.

In sum, this new model contributes to improve the project management in a game development process, now covering the entire process, since the creative parts until the productive

ones. In a company vision, the DT can contribute to a strategy of product differentiation and also can serve to more fully understand game requirements, while agile development practices can increase development efficiency, thus, the use of both approaches could be considered as two competitive advantages to a game company.

Limitations

The results seen in this research cannot be generalized to the entire Brazilian and world market. However, the proposed model was based on field evidence, as professionals from five companies were interviewed and their contribution can certainly be expanded.

Futures researches

Finally, three opportunities for future research can be indicated: first, broadening the respondents' base, which will allow the possibility to verify the generalization of the results found in this research. Secondly, conducting an action research, focusing on the combined use of DT with AMs, and identifying strengths and weaknesses of the use of the integrated approach. The third one is to expand research in DT literature to find more techniques that would fit into the development of games, being the choice of the best based on opinions of developers or experts of the games area.

REFERENCES

(PMI), P. M. (2008). *A Guide to the project management body of knowledge - PMBOK - 4ª* ed. Pennsylvania: Project Management Institute.

Abrahamsson, P., Conboy, K., & Wang, X.. (2009). "'Lots done, more to do': the current state of agile systems development research". European Journal of Information Systems 18(4):281–84.

Aldridge, M. (1980). New Design Thinking sets Roll Formers Future. Metalworking Production, pp. 124-127.

Alliance, A. (2001). Principles: the agile alliance. Acesso em 2016, disponível em Agile Manifesto: http://agilemanifesto.org/principles.html

Akta , A., & Orcun, E. (2014). A survey of computer game development. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 1548512914554405-. http://doi.org/10.1177/1548512914554405

Albino, R. D., Souza, C. A. de, & Prado, E. P. V. (2014). Benefícios Alcançados por Meio de um Modelo de Gestão Ágil de Projetos em uma Empresa de Jogos. *Revista de Gestão e Projetos - GeP*, *5*, 15–27.

Bauer, R., & Eagen, W. (2008). Design thinking: Epistemic plurality in management and organization. *Aesthesis: International Journal of Art and Aesthetics in Management and Organizational Life*, *2*(3), 568–596.

Berthêm, A. C. (2007). *Desenvolvimento de Jogos Eletrônicos: Teoria e Prática - 2ª ed.* São Paulo: Novatec.

Boehm, B. (2002). Get Ready for Agle Methods, with Care. *IEEE Computer*, *1*(35), 64–69. http://doi.org/10.1109/2.976920

Bonini, L. A., & Sbragia, R. (2011). O modelo de design thinking como indutor da inovação nas empresas: um estudo empírico. *Revista de Gestão e Projetos*, *2*(1), 03–25. http://doi.org/10.5585/gep.v2i1.36

Brathwaite, B., & Schreiber, I. (2009). *Challenges for game designers.* Boston, MA: Charles River Media.

Brown, T. (2008). Design Thinking. *Harvard Business Review*. http://doi.org/10.1145/2535915

Brown, T. (2009). *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*. *Director 00123242* (Vol. 31). http://doi.org/10.1017/CBO9781107415324.004

Brown, T., & Katz, B. (2011). Change by design. *Journal of Product Innovation Management*, *28*(3), 381–383.

Buchanan, R. (1992). Wicked Problems in Design Thinking. *The MIT Press*, *8*(2), 5–21.

Buchanan, R. (2014). in Design Thinking Wicked Problems, *8*(2), 5–21.

Buchanan, R. (2016). Wicked Problems in Design Thinking Author (s): Richard Buchanan Published by : MIT Press Stable URL : http://www.jstor.org/stable/1511637 Accessed : 01-03-2016 00 : 56 UTC Your use of the JSTOR archive indicates your acceptance of the Terms & Conditions, *8*(2), 5–21.

Cao, L., Mohan, K., Xu, P., & Ramesh, B. (2009). A

framework for adapting agile development methodologies. *European Journal of Information Systems*, *18*(4), 332–343. http://doi.org/http://dx.doi.org/10.1057/ejis.2009.26

Cockburn, A. (2007). *Agile Software Development: The Cooperative Game.* Addison-Wesley.

Cockburn, Alistair e Jim Highsmith. (2001). "Agile Software Development:The People Factor". Computer 34(11):131–33.

Conboy, K. (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information Systems Research*, *20*(3), 329–354.

Cooper, R. G. (2014). Invited Article: What's Next?: After Stage-Gate. *Research-Technology Management*, *57*(1), 20–31. http://doi.org/10.5437/08956308X5606963

Cross, N. (1982). Designerly ways of knowing. *Design*, *3*(4), 221–227.

Cross, N. (2006). *Designerly Ways of Knowing.* Berlim: Springer.d.school. (2008). *Bootcamp Bootleg.* Acesso em 25 de Fevereiro de 2016, disponível em d.school: Institute of Design at Stanford: <http://dschool. stanford.edu/wp-content/uploads/2011/ 03/METHODCARDS2010v6.pdf>

Dingsoyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, *85*(6), 1213–1221. http://doi.org/10.1016/j.jss.2012.02.033

Dorst, C. H. (2006). Design Problems and Design Paradoxes. *Design Issues*, *22*(3), 4–17. http://doi.org/10.1162/desi.2006.22.3.4

Dorst, K. (2011). The core of "design thinking" and its application. *Design Studies*, *32*(6), 521–532. http://doi.org/10.1016/j.destud.2011.07.006

Dunne, D., & Martin, R. (2006). Design thinking and how it will change management education: An interview and discussion. *Academy of Management Learning and Education*, *5*(4), 512–523. http://doi.org/10.5465/AMLE.2006.23473212

Erdogmus, H., Morisio, M., & Torchiano, M. (2005). NRC Publications Archive Archives des publications du CNRC. *IEEE Transactions on Software Engineering*, *31*(1), 1–12. http://doi.org/10.1039/B910216G

Fleury, A. L., Stabile, H., & Carvalho, M. M. De. (2016). An Overview of the Literature on Design Thinking : Trends and Contributions *. *International Journal of Engineering Education*, *32*(4), 1704–1718.

Henderson-Sellers, B., & Serour, M. K. (2005). Creating a Dual-Agility Method: The Value of Method Engineering. *Journal of Database Management*, *16*(4), 1–23. Recuperado de http://search.proquest.com/docview/199609160?accountid=14719%5Cnhttp://openurl.uquebec.ca:9003/uqam?url_ver=Z39.88-2004&rft_val_fmt=info:ofi/fmt:kev:mtx:journal&genre=article&sid=ProQ:ProQ:abiglobal&atitle=Creating+a+Dual-Agility+Method:+The+Value+of+Met

Highsmith, J., Consortium, C., & Cockburn, A. (2001). Development : The Business of Innovation. *IEEE Computer*, 120–127.

Hodgson, D., & Briand, L. (2013). Controlling the uncontrollable: "Agile" teams and illusions of autonomy in creative work. *Work Employment and Society*, *27*(2), 308–325. http://doi.org/10.1177/0950017012460315

IDEO. (2011). Human Centered Design - 2ª ed. Johansson-Skoldberg, U. et al. (2013). "Design Thinking : Past , Present and Possible Futures". Creativity and Innovation Management 22(2):121–46.

Kimbell, L. (2011). Rethinking Design Thinking: Part I. *Design and Culture*, *3*(3), 285–306. http://doi.org/10.2752/175470811X13071166525216

Kimbell, L. (2012). Rethinking Design Thinking: Part II. *Design and Culture*, *4*(2), 129–148. http://doi.org/10.2752/175470812X13281948975413

Lalsing, V. (2012). "People Factors in Agile Software Development and Project Management". International Journal of Software Engineering & Applications 3(1):117–37. Recuperado (http://airccse.org/journal/ijsea/papers/3112ijsea09.pdf).

Larsen, L.J., Majgaard, G., 2016. Expanding the Game Design Space – Teaching Computer Game Design in Higher Education. Designs for Learning 8, 13–22. doi:10.16993/dfl.68

Lee, G., & Xia, W. (2010). Toward agile: An integrated analysis of quantitative and qualitative field data on software development agility. *MIS Quarterly*, *34*(1), 87–114. http://doi.org/Article

Liem, A. & Brangier, E. (2012). "Innovation and design approaches within prospective ergonomics". IOS Press 41(SUPPL.1):5243–50.

Lockwood, T. (2009). Frameworks of Design Thinking. *Design Management Journal*, *4*(1), 3. http://doi.org/10.1111/j.1942-5074.2009.00001.x

Martin, R. 2009. The Design of Business: Why Design Thinking Is the Next Competitive Advantage. Cambridge MA: Harvard Business Press.

Manninen, T., & Kujanpa, T. V. (2006). *Game production process, a preliminary study (ELIAS project reports), Ludo-Craft, ELIAS-project (European Union Interreg III A Karjala), University of Oulu.* Finlândia.Martin, R. (2009). *The Design of Business: Why Design Thinking is the Next Competitive Advantage*. *Harvard Business Press*. Recuperado de http://www.amazon.ca/Design-Business-Thinking-Competitive-Advantage/dp/1422177807

McCarthy, D. C., & Byron, S. (2005). *The complete guide to game development, Art, and Design.* East Sussex, UK.: Ilex.

Moe NB, Dingsøyr T, D. T. (2009). Overcoming Barriers to Self-Management in Software Teams. *IEEE Software*.

Nerur, S., Mahapatra, R., & Mangalaraj, G. (2005). "Challenges of Migrating to Agile Methodologies". Communications of the ACM 48(2):72–78.

Razzouk, R., & Shute, V. (2012). What Is Design Thinking and Why Is It Important? *Review of Educational Research*, *82*(3), 330–348. http://doi.org/10.3102/0034654312457429

Research, S. (December de 2016). *Market brief year in review.* Acesso em April de 2017, disponível em SuperData: https://www.superdataresearch.com/market-data/market-brief-year-in-review/

Rittel, H. W. J., & Webber, M. M. (1973). Dilemmas in a general theory of planning. *Policy Sciences*, *4*(2), 155–169. http://doi.org/10.1007/BF01405730

Schwaber, K. (2004). *Agile Project Management with SCRUM.* Microsoft Press.

Siau, Keng. (2005). "A Retrospective Review of JDM from 2003 to 2005 and a Discussion on Publication Emphasis of JDM for the Next Two to Three Years". Journal of Database Management 16(4):58–71.

Simon, H. A. (1973). The structure of ill structured problems. *Artificial Intelligence*, *4*(3–4), 181–201. http://doi.org/10.1016/0004-3702(73)90011-8

Sloper, T. (2002). Following Up After the Game is Released: It's not Over when it's Over. *Game Design Perspectives*.

Sommerville, I. (2003). *Engenharia de Software - 6ª ed.* São Paulo: Addison Wesley.

Stacey, R. D., Griffin, D., & Shawn, P. (2002). Complexity and Management: Fad or Radical Challenge to Systems Thinking? *Journal of Macromarketing*, *22*(2), 198–201. http://doi.org/10.1177/0276146702238228

Seidel, V. P. & Fixson, S. K. (2013). "Adopting design thinking in novice multidisciplinary teams: The application and limits of design methods and reflexive practices". Journal of Product Innovation Management 30(SUPPL 1):19–33.

Vetterli, C., Brenner, W., Uebernickel, F., & Petrie, C. 2013. "From palaces to yurts: Why requirements engineering needs design thinking". IEEE Internet Computing 17(2):91–94.

Williams, L., Kessler, R. R., Cunningham, & W., Jeffries, R. (2000). Strengthening the case for pair programming. *IEEE Software*, *17*, 19–25. http://doi.org/10.1016/S0140-6736(05)74076-9