

UNIVERSIDADE FEDERAL DO PAMPA

Murilo da Rosa Bianchin

**AgileGD: Um Processo Inspirado em  
Abordagens Ágeis para o Desenvolvimento  
de Jogos**

Alegrete  
2019



Murilo da Rosa Bianchin

# AgileGD: Um Processo Inspirado em Abordagens Ágeis para o Desenvolvimento de Jogos

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Software da Universidade Federal do Pampa como requisito parcial para a obtenção do título de Bacharel em Engenharia de Software.

Orientador: Prof. Dr. João Pablo Silva da Silva

Coorientador: Prof. Me. Jean Felipe Patikowski Cheiran

Alegrete  
2019

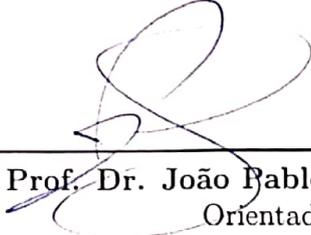


Murilo da Rosa Bianchin

## AgileGD: Um Processo Inspirado em Abordagens Ágeis para o Desenvolvimento de Jogos

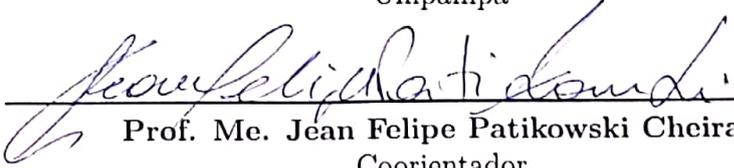
Trabalho de Conclusão de Curso apresentado  
ao Curso de Engenharia de Software da Uni-  
versidade Federal do Pampa como requisito  
parcial para a obtenção do título de Bacharel  
em Engenharia de Software.

Trabalho de Conclusão de Curso defendido e aprovado em 26 de Junho de 2019  
Banca examinadora:



---

Prof. Dr. João Pablo Silva da Silva  
Orientador  
Unipampa



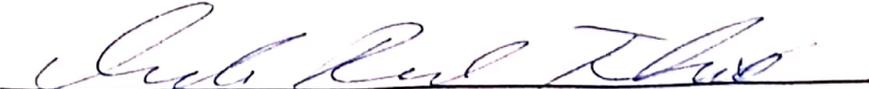
---

Prof. Me. Jean Felipe Patikowski Cheiran  
Coorientador  
Unipampa

Andrea P. Bordin

---

Profa. Dra. Andrea Sabedra Bordin  
Unipampa



---

Prof. Dr. Marcelo Resende Thielo  
Unipampa



## **AGRADECIMENTOS**

Primeiramente, agradeço a minha família, meus pais e minha irmã pelo apoio e carinho de sempre, minha namorada e os meus sogros que nunca se negaram me ajudar diante de qualquer situação. Sem vocês não sou nada, não chegaria onde cheguei e não me tornaria a pessoa que sou hoje.

Agradeço ao meu orientador João Pablo Silva da Silva e ao meu coorientador Jean Felipe Patikowski Cheiran, nos quais nunca mediram esforços para me auxiliar no decorrer do curso e no presente trabalho. Pessoas ímpares, que sempre repassaram uma extrema educação, conhecimento e competência, prezando sempre pelo sucesso e educação de qualidade de todos os seus alunos.

Além disso, também agradeço aos grandes amigos que esse curso me apresentou: Neto, João e Alex. Obrigado por sempre me ajudarem e me motivarem em todos os momentos difíceis, vocês foram fundamentais no meu crescimento e jornada dentro do curso.

Por fim, gostaria de agradecer a todos que de alguma forma contribuíram de forma indireta e/ou direta para a realização deste trabalho, sejam amigos ou familiares, muito obrigado pelo auxílio!



## RESUMO

A adoção de processos de software é uma necessidade no desenvolvimento de sistemas. Apesar da vasta quantidade de processos que auxiliam no desenvolvimento tradicional de software, quando o foco é o desenvolvimento de jogos a situação muda. Além de serem poucos os processos formalmente descritos que atendem às especificidades da criação de um jogo, os desenvolvedores de jogos ainda apresentam certa resistência na adoção de alguns desses processos. Em grandes produtoras de jogos, variações dos processos descritos na literatura são adotadas por se adaptarem melhor ao perfil interdisciplinar das equipes de desenvolvimento, porém muitas vezes as necessidades específicas desse tipo de projeto levam à adoção de variações *ad hoc*, levando à incerteza da sua efetividade. Este trabalho tem como objetivo propor um processo inspirado em abordagens ágeis e adaptável para diferentes contextos na área de jogos chamado AgileGD, tendo em vista que os processos específicos para o desenvolvimento de jogos não são adaptáveis aos diferentes tipos de equipes ou acabam tendo outras limitações que comprometem a qualidade do produto final. Para identificar quais metodologias ágeis estão sendo utilizadas no desenvolvimento de jogos, realizamos nesse trabalho uma Revisão Sistemática na Literatura, a fim de identificar quais metodologias ágeis são mais usadas, o porquê delas serem adaptadas e o que foi adaptado na estrutura. Após a revisão sistemática começamos a elaborar o processo com o objetivo de seguir alguns requisitos levantados durante a análise dos dados provenientes dos trabalhos relacionados, sua modelagem gráfica foi realizada utilizando o Business Process Model and Notation (BPMN). Por fim, o processo foi validado através de um Grupo Focal com estudantes que possuem experiência na área de desenvolvimento de jogos e na utilização de métodos ágeis. Após a execução do Grupo Focal os dados foram analisados e foi identificado muito *feedback* positivo e pouco relato de críticas ou mudanças mais acentuadas no processo. Avaliamos os relatos dos participantes à luz do desenvolvimento de jogos, incorporando características que julgamos necessárias, descartando características que não se encaixam na estrutura do processo e mantendo o registro das características na zona de incerteza para que o leitor possa julgar sua relevância. O processo apresentado neste trabalho possui evidências que ele atingiu os seus objetivos, otimizando o trabalho da equipe no desenvolvimento de jogo, oferecendo o suporte de processo necessário para a equipe e respeitando os requisitos definidos.

**Palavras-chave:** Processo de Software. Metodologia Ágil. Jogos.



## ABSTRACT

The adoption of software processes is a need in systems development. Although there is a huge amount of processes to guide the traditional software development, this situation changes when related to game development. Besides the lack of processes formally described that meet the needs of a game development, game developers still present resistance in adopt the existent processes. In big game companies, variations of processes described in the literature are used by developers, once these adapted processes fit better to the interdisciplinary profile of the development team. However, the specific needs of these projects often lead to the adoption of ad hoc variations, leading to the uncertainty of the project effectiveness. This study aims at the proposal of AgileGD process, a process inspired by agile approaches and adaptable to different contexts in the game development, considering that specific processes used for game development are not adaptable to different types of teams or they have other limitations that compromise the quality of the final product. In order to identify which agile methodologies are being used in the development of games, we perform a Systematic Literature Review in order to figure out which agile methodologies are most used, why they are adapted and what has been adapted in the structure. After the systematic review, we began to elaborate the process with the objective of following some requirements raised during the analysis of the data coming from the related works. It was modeled in Business Process Model and Notation (BPMN). Finally, the process was validated through a Focus Group with students who have experience in the area of game development and the use of agile methods. After soon after the implementation of the Focus Group the data were analyzed and positive feedback was identified with few reports of criticism or changes in the process. We evaluate the participants' reports in the light of game development, incorporating features that we deem necessary, discarding features that do not fit the process structure and keeping record of features in the zone of uncertainty so that the reader can judge their relevance. The process presented in this work has evidences that it has achieved its objectives, optimizing the work of the team in game development, offering the necessary process support to the team and following the defined requirements.

**Key-words:** Software Process. Agile Methodology. Games.



## LISTA DE FIGURAS

Figura 1 – Estrutura da <i>string</i> de busca. . . . .	30
Figura 2 – Estrutura do AgileGD. . . . .	40
Figura 3 – Processo AgileGD completo em alto nível. . . . .	40
Figura 4 – Subprocesso Conceber Jogo e suas atividades internas. . . . .	41
Figura 5 – Subprocesso Realizar Desenvolvimento do Jogo (Sprint) e suas atividades internas. . . . .	42
Figura 6 – Subprocesso Atividade Diária e suas atividades internas. . . . .	43
Figura 7 – Subprocesso Realizar Tarefas Diárias e suas atividades internas . . . .	43
Figura 8 – Subprocesso Codificar Funcionalidades do Jogo. . . . .	44
Figura 9 – Subprocesso Pré-Lançamento do Jogo e suas atividades internas. . . . .	46
Figura 10 – Gráficos de Pizza referentes as respostas do problema 1. . . . .	53
Figura 11 – Gráficos de Pizza referentes as respostas do problema 2. . . . .	54
Figura 12 – Gráficos de Pizza referentes as respostas do problema 3. . . . .	56
Figura 13 – Gráficos de Pizza referentes as respostas do problema 4. . . . .	57



## LISTA DE TABELAS

Tabela 1 – Metodologias ágeis utilizadas nos trabalhos relacionados. . . . .	36
Tabela 2 – Motivos para adaptar o Scrum relatados em cada trabalho. . . . .	36
Tabela 3 – Papéis, atividades e artefatos adaptados do Scrum Tradicional. . . . .	37



## **LISTA DE SIGLAS**

**AAGDM** Agent Agile Game Development Methodology

**AOAB** Agent Oriented Agile Based

**AOSE** Agent Oriented Software Engineering

**ASD** Adaptive Software Development

**BPMN** Business Process Model and Notation

**DSDM** Dynamic Systems Development Method

**GDD** Game Design Document

**GUP** Game Unified Process

**GWP** Game Waterfall Process

**IGDD** Improved Game Design Document

**RUP** Rational Unified Process

**TCLE** Termo de Consentimento Livre e Esclarecido

**TDD** Test Driven Development

**UP** Unified Process

**XGD** Extreme Gaming Development

**XP** Extreme Programming



## SUMÁRIO

1	INTRODUÇÃO . . . . .	19
1.1	Motivação . . . . .	20
1.2	Objetivos . . . . .	20
1.3	Metodologia . . . . .	20
1.4	Contribuições . . . . .	21
1.5	Organização . . . . .	21
2	FUNDAMENTAÇÃO TEÓRICA . . . . .	23
2.1	Desenvolvimento de Jogos Eletrônicos . . . . .	23
2.2	Processo de Software . . . . .	23
2.2.1	Modelagem de Processo . . . . .	24
2.3	Metodologias Ágeis . . . . .	25
2.3.1	Scrum . . . . .	25
2.3.2	Extreme Programming . . . . .	26
2.4	Lições do Capítulo . . . . .	27
3	TRABALHOS RELACIONADOS . . . . .	29
3.1	Metodologia da Revisão Sistemática . . . . .	29
3.1.1	Questões de Pesquisa . . . . .	29
3.1.2	Processo de Busca . . . . .	29
3.1.3	Processo de Seleção . . . . .	30
3.1.4	Coleta e Análise de Dados . . . . .	31
3.2	Resultados . . . . .	32
3.3	Respostas para as Questões de Pesquisa . . . . .	35
3.3.1	Quais são as metodologias ágeis utilizadas para desenvolver jogos e qual é a mais utilizada? . . . . .	35
3.3.2	Por que foram customizadas as metodologias ágeis para o desenvolvimento de jogos? . . . . .	36
3.3.3	Que papéis, artefatos e atividades foram adaptados nas metodologias ágeis? . . . . .	37
3.4	Lições do Capítulo . . . . .	38
4	PROPOSTA DE PROCESSO ÁGIL PARA JOGOS . . . . .	39
4.1	Visão Geral do AgileGD . . . . .	39
4.2	Especificação do AgileGD . . . . .	40
4.2.1	Fase de Concepção . . . . .	41
4.2.2	Fase de Desenvolvimento . . . . .	42
4.2.3	Fase de Transição . . . . .	46
4.3	Lições do Capítulo . . . . .	46

5	AVALIAÇÃO DO AGILEGD . . . . .	49
5.1	Planejamento . . . . .	49
5.2	Execução . . . . .	50
5.2.1	Resultados . . . . .	52
5.2.1.1	Problema 1 . . . . .	52
5.2.1.2	Problema 2 . . . . .	53
5.2.1.3	Problema 3 . . . . .	55
5.2.1.4	Problema 4 . . . . .	56
5.2.2	Ameaças . . . . .	58
5.3	Lições do Capítulo . . . . .	58
6	CONSIDERAÇÕES FINAIS . . . . .	61
6.1	Trabalhos Futuros . . . . .	61
	REFERÊNCIAS . . . . .	63
	APÊNDICES . . . . .	67
	APÊNDICE A – MATERIAL DE APOIO AO GRUPO FO- CAL . . . . .	69
	APÊNDICE B – GUIA DE ADAPTAÇÃO . . . . .	83

## 1 INTRODUÇÃO

Com o passar dos anos, a indústria desenvolvimento de jogos vem crescendo de uma forma que ninguém imaginou. De acordo com Rabin (2012) os jogos ainda não alcançaram o *status* de filmes, mas isto está mudando de forma rápida. As habilidades e os talentos artísticos envolvidos na produção de jogos logo irão rivalizar com os de filmes, com produções típicas de jogos alcançando orçamentos de \$10 a \$20 milhões de dólares e sem previsão de limite. Por trás disso, existem várias equipes de desenvolvedores de *games* de diferentes tamanhos e orçamentos, embora todos tenham uma busca em comum que é a produção de jogos que farão sucesso entre os usuários que estão cada vez mais exigentes na busca do entretenimento.

*Games* podem ser produzidos em casa por desenvolvedores *indie*<sup>1</sup> com poucos ou nenhum recurso ou por grandes empresas que possuem um grande orçamento. Contudo, não é necessariamente se precisa investir muito para produzir um jogo de qualidade. Um exemplo disso é o jogo Minecraft, que foi desenvolvido por um desenvolvedor *indie* e foi vendido em 15 de setembro de 2014 para a empresa multinacional Microsoft por 2,5 bilhões de dólares e sendo acessado por mais de 11 milhões de jogadores. Segundo Persson (2011) citado por Duncan (2011), o desenvolvedor do jogo Minecraft, Markus “Notch” Persson, afirmou “[O modelo] Cascata está morto, vida longa ao [método] ágil!” [tradução do autor] referindo-se à metodologia ágil de desenvolvimento usada neste jogo.

A motivação para a afirmação de “Notch” é que, os métodos ágeis podem fornecer práticas simples para as equipes e que agilizem o processo de desenvolvimento do jogo, mas sem perder a qualidade, diminuindo o retrabalho e mantendo uma relação próxima dos integrantes da equipe, ao contrário dos métodos tradicionais que são mais burocráticos com as suas documentações e se preocupam menos com as relações humanas.

Portanto, não somente para a criação de jogos, como também para o desenvolvimento de software, deve-se ter uma grande atenção na escolha de processos adequados, com o objetivo de auxiliar os desenvolvedores na criação de produtos com qualidade. Em vista disso, os processos usados para desenvolver jogos foram criados a partir dos processos de software já existentes, adaptando-os a fim de compatibilizar as diferentes etapas de desenvolvimento e sem perder a essência dos processos originais. Segundo Rodrigues, Machado e Valença (2010), entre os processos específicos para o desenvolvimento de jogos se pode destacar o Game Unified Process (GUP), o Extreme Gaming Development (XGD) e o Game Waterfall Process (GWP). Além disso, de acordo com Petrillo e Pimenta (2010), apesar da existência das metodologias ágeis, os processos mais utilizados no mercado de desenvolvimento de jogos ainda são GWP e GUP. Isso ocorre porque os dois processos são adaptações de processos mais tradicionais na indústria de desenvolvimento de software e, por isso, há perpetuação da resistência na adoção de práticas ágeis que ainda são vistas

---

<sup>1</sup> Jogos *Indies* são jogos eletrônicos criados por uma pessoa ou pequenas equipes com ou sem apoio financeiro de publicadoras de jogos eletrônicos.

por alguns gestores e instituições, infelizmente, como indisciplinadas.

## 1.1 Motivação

A partir da realização de uma pesquisa, buscando encontrar processos ágeis de desenvolvimento de jogos, inclusive analisando *post mortem* de jogos bem sucedidos, constatou-se que os processos estão sendo adaptados para o escopo de cada projeto e que são poucos os trabalhos que apresentam a estrutura completa do seu processo na literatura acadêmica. Com isso foram identificadas algumas metodologias ágeis adaptadas, porém todas foram adaptadas para que o processo ágil atenda o escopo daquela equipe e projeto específicos.

Segundo Petrillo (2008) depois de analisar inúmeros *postmortens* de desenvolvimento de jogos, os autores afirmam que um problema que pode ser destacado como específico da indústria de jogos é a comunicação na equipe, uma vez que aparentam dividir suas equipes em “os artistas” e “os programadores” quando utilizados processos tradicionais. Por causa desses problemas as equipes acabam se desmotivando e diminuindo sua comunicação, assim, impactando de alguma forma no resultado final do produto.

Já Liana et al. (2014) diz que uma das principais barreiras encontradas durante a criação de um jogo é a falta de processos específicos para jogos que auxiliem as diferentes equipes no desenvolvimento de jogos.

## 1.2 Objetivos

Nós definimos como objetivo geral deste trabalho um processo de desenvolvimento baseado em princípios ágeis para desenvolvimento de jogos digitais. A seguir, apresentamos os objetivos específicos deste trabalho:

- estabelecimento de uma base conceitual e tecnológica relacionada ao objetivo do trabalho;
- investigação das metodologias ágeis propostas para o desenvolvimento de jogos;
- especificação do processo de desenvolvimento aderente ao manifesto ágil;
- avaliação através de um estudo empírico.

## 1.3 Metodologia

Para propormos o processo identificamos a necessidade de saber quais metodologias ágeis estão sendo utilizadas para o desenvolvimento de jogos, então realizamos uma Revisão Sistemática da Literatura nos baseando em O’Hagan, Coleman e O’Connor (2014). Através da Revisão Sistemática pesquisamos artigos importantes para o escopo do

estudo, aplicando critérios de inclusão, exclusão e de qualidade com o objetivo de selecionar os artigos que realmente se encaixam no escopo desta pesquisa. Através da separação dos estudos mais relevantes e da síntese dos dados retirados, foi possível identificar quais metodologias ágeis estão sendo mais utilizadas e detalhes sobre as suas estruturas. Após serem analisados esses dados, foi possível elaborar uma série de requisitos que o processo proposto neste trabalho segue. Na sequência estruturamos e modelamos o processo chamado AgileGD, respeitando os requisitos definidos. Contornando problemas e dificuldades encontrados durante a sua elaboração e considerando o objetivo do processo sempre seguir os requisitos estruturais definidos, através disso acreditamos que o processo atende o que foi planejado. Por fim, foi realizado um Grupo Focal seguindo o guia de Kontio, Bragge e Lehtola (2008) como forma de validar o processo proposto. Através da apresentação dos problemas encontrados durante a elaboração do processo e das soluções que sugerimos, foi possível obter diferentes opiniões relacionadas ao processo e com isso o refinamos com alterações julgadas por nós como necessárias, não necessárias ou em análise.

#### 1.4 Contribuições

Este trabalho tem como principal contribuição apresentar a elaboração de um processo de desenvolvimento específico para o desenvolvimento de jogos, com os problemas enfrentados e como foram solucionados para que o processo seguisse os requisitos previamente definidos. Outras contribuições deste trabalho são:

- apresentação do estado da arte a partir de uma revisão sistemática na literatura;
- apresentação da validação do processo a partir de o grupo focal.

#### 1.5 Organização

O restante desta monografia está organizada como segue. No Capítulo 2, apresentamos a fundamentação teórica, onde são abordados o desenvolvimento de jogos, processos de software e as metodologias ágeis. No Capítulo 3, apresentamos os trabalhos relacionados, onde são abordados os trabalhos acadêmicos, correspondentes à pesquisa realizada neste trabalho. No Capítulo 4, apresentamos o andamento do trabalho, onde são abordados a estratégia de avaliação, a proposta da metodologia ágil e os materiais e métodos utilizados. No Capítulo 5, apresentamos a estratégia de avaliação utilizadas nesse estudo e seus resultados. Por fim, no Capítulo 6 apresentamos as considerações finais desse trabalho.



## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo embasamos os aspectos teóricos e tecnológicos deste trabalho. Na Seção 2.1, apresentamos a definição do desenvolvimento de jogos eletrônicos, onde são evidenciadas as suas características e curiosidades. Na Seção 2.2 definimos processo de software e caracterizamos a modelagem de processo. Por fim, na Seção 2.3 desenvolvemos aspectos relacionados aos métodos ágeis com ênfase em Scrum e Extreme Programming (XP).

### 2.1 Desenvolvimento de Jogos Eletrônicos

Segundo Rabin (2012), Steve Russel criou o jogo Spacewar que inspirou a indústria multibilionária de jogos. O jogo foi desenvolvido no DEC PDP-1 que era um computador que ocupava uma mesa inteira, executado em um período um curso de seis meses e quase 200 horas em 1961 no Massachusetts Institute of Technology. Porém, na época não existiam computadores caseiros, e os criadores de Spacewar não pensavam em ganhar dinheiro com a invenção.

Anos se passaram e a indústria dos jogos está crescendo. Atualmente, há jogos que atendem a públicos diversos, em relação a faixa etária, gênero, nacionalidade, cultura, situação socioeconômica, etc, que estão disponíveis em celulares, *tablets*, computadores, *videogames* entre outros dispositivos eletrônicos. Os jogos estão também chegando a valores de produção e lucros acima do imaginado. Um exemplo disso é o jogo Grand Theft Auto V que superou outros lançamentos de outras mídias muito populares da mesma época, como, por exemplo, os filmes Star Wars e Harry Potter, para se tornar o título de mídia mais lucrativo já lançado (TAMBURRO, 2018).

O desenvolvimento de jogos depende de uma equipe de desenvolvimento, que hoje em dia pode ser financiada por uma produtora com significativo investimento econômico, que são os casos dos jogos AAA<sup>1</sup>. Rabin (2012) define os jogos AAA como jogos com uma qualidade superior e nos quais se tem um considerável investimento econômico e de esforços de produção, visando excelência técnica na produção. O inverso desse cenário são os jogos *indie*, que segundo Novak (2010) são produzidos por uma pessoa ou uma equipe pequena, com pouco ou nenhum suporte financeiro e, principalmente, sem o suporte de uma produtora de *games*.

### 2.2 Processo de Software

Segundo Wazlawick (2013) o processo é um conjunto de regras que definem como um projeto deve ser executado. Segundo ele, o processo normalmente é utilizado por uma

---

<sup>1</sup> “A lot of time, A lot of money, A lot of resources” (ou muito tempo, muito dinheiro, muitos recursos em uma tradução livre).

empresa como um conjunto de regras específicas que seus respectivos funcionários devem seguir em cada projeto.

Pressman (2011) por sua vez define o processo como um conjunto de atividades, ações e tarefas realizadas na criação de algum produto de trabalho. Ele ainda diz que o processo é uma abordagem adaptável que possibilita a equipe de software realizar o trabalho de selecionar e escolher o conjunto apropriado de ações e tarefas.

Sommerville, Arakaki e Melnikoff (2008) abordam o processo da seguinte forma “Um processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software”. Dessa forma o processo de software pode ser visto como atividades que se utilizam de métodos, ferramentas e práticas necessárias para a construção de um software. De acordo com Sommerville, Arakaki e Melnikoff (2008), existem muitos processos de software diferentes, porém todos devem incluir quatro atividades fundamentais para a engenharia de software:

- especificação de software: a funcionalidade do software e as restrições de seu funcionamento devem ser definidas;
- projeto e implementação de software: o software deve ser produzido para atender às especificações;
- validação de software: o software deve ser validado para garantir que atenda às demandas do cliente;
- evolução de software: o software deve evoluir para atender às necessidades de mudança dos clientes.

### 2.2.1 Modelagem de Processo

Segundo Pressman (2011) um modelo de processo de software é uma representação abstrata de um processo de software. Cada modelo de processo representa um processo sob determinada perspectiva e, dessa forma, fornece somente informações parciais sobre esse processo.

Uma forma de modelar graficamente os processos de software é utilizando o BPMN. Segundo White (2006) a notação para a modelagem de processo de negócios BPMN é uma notação padrão para o desenho de fluxogramas em processos de negócios. Seu principal objetivo é prover uma notação com interface amigável e que seja compreendida por todos os usuários envolvidos, desde os analistas de negócios até os analistas de sistemas.

Briol (2008) diz que a modelagem em BPMN é realizada através de diagramas simples, com um conjunto pequeno de elementos gráficos. Os elementos são agrupados em quatro categorias básicas, são elas: *Flow Objects*, *Connecting Objects*, *Swim Lanes* e *Artefacts*.

## 2.3 Metodologias Ágeis

No atual cenário de desenvolvimento de software as empresas precisam trabalhar a fim de ter um retorno rápido às constantes mudanças em relação à tecnologia, ao mercado e principalmente aos requisitos de software. Essas mudanças fazem com que o software em desenvolvimento sofra com requisitos cada vez mais instáveis. Por conta desses problemas, surgiu o desenvolvimento ágil, que antigamente era chamado de métodos leves. Desde então vários métodos ágeis foram propostos, porém todos com a mesma ideologia, representada no manifesto ágil, o qual valoriza: (BECK et al., 2001).

- indivíduos e interação entre eles mais que processos e ferramentas;
- software funcionando mais que documentação abrangente;
- colaboração com cliente mais que negociação de contratos;
- responder a mudanças mais que seguir um plano.

Segundo Pressman (2011), os métodos ágeis se desenvolveram em um esforço para sanar fraquezas reais e perceptíveis da engenharia de software convencional. O modelo mais utilizado dentre todos os modelos de processos ágeis é o XP seguido pelos métodos Adaptive Software Development (ASD), Scrum, Dynamic Systems Development Method (DSDM) e Crystal.

Para Wazlawick (2013) “Os modelos ágeis de desenvolvimento de software seguem uma filosofia diferente da filosofia dos modelos prescritivos”. Em vez de apresentar uma “receita de bolo”, com fases ou tarefas a serem executadas, eles focam em valores humanos e sociais.

### 2.3.1 Scrum

O Scrum foi criado por dois dos profissionais signatários do manifesto ágil, Ken Schwaber e Jeff Sutherland. Segundo Schwaber e Sutherland (2016) o Scrum é “um *framework* para desenvolver e manter produtos complexos”.

Pressman (2011) por sua vez diz que o Scrum é um método de desenvolvimento ágil de software, que os princípios do Scrum são consistentes com o manifesto ágil usados para orientar as atividades de desenvolvimento dentro de um processo. O autor diz também que as equipes possuem papéis bem definidos, são eles:

- *Scrum Master*: responsável por inspirar os demais membros do time a serem auto-gerenciáveis e garantir que todos cumpram as regras de Scrum;
- *Product Owner*: representa o papel de todos os clientes do produto;
- Time de Desenvolvimento: responsáveis pelo desenvolvimento dos requisitos.

O Scrum é realizado a partir de iterações chamadas de *Sprints*. Essas iterações possuem duração em média de 2 a 4 semanas, sendo recomendado não exceder esse período. No Scrum existem também artefatos, tais como:

- *Product Backlog*: artefato que contém todos os requisitos especificados para o produto;
- *Sprint Backlog*: artefato que contém os requisitos a serem desenvolvidos durante o *Sprint* atual.

De acordo com Sommerville, Arakaki e Melnikoff (2008) o *Sprint* é como uma unidade de planejamento na qual o trabalho a ser feito é avaliado, os recursos para o desenvolvimento são selecionados e o software é implementado. Ainda assim, vale ressaltar que antes do início de cada *Sprint* é realizada uma reunião chamada de Reunião de Planejamento do *Sprint*, na qual é decidido quais requisitos do *Product Backlog* irão compor o *Sprint Backlog* naquele determinado *Sprint*.

No Scrum existem vários eventos que ocorrem durante os *Sprints*:

- Reunião Diária: que é uma breve reunião com o intuito de acompanhar o projeto, na qual os membros da equipe relatam o que foi feito no dia e o que será feito no próximo dia;
- Reunião de Revisão do *Sprint*: ao fim de cada *Sprint*, é realizada uma reunião de revisão, na qual é apresentado ao *Product Owner* o que foi desenvolvido durante aquele *Sprint*;
- Reunião de Retrospectiva do *Sprint*: que tem como objetivo avaliar os pontos fortes e fracos que foram observados durante o *Sprint*, além de apontar as melhorias a fim de corrigir os pontos fracos.

### 2.3.2 Extreme Programming

A metodologia ágil XP foi criada por Kent Beck que foi um engenheiro de software americano criador também do Test Driven Development (TDD). Segundo Sommerville, Arakaki e Melnikoff (2008) o XP é talvez um dos métodos ágeis mais conhecidos e mais utilizados. O autor diz também que o XP foi desenvolvido para impulsionar práticas reconhecidamente boas, como o desenvolvimento iterativo.

O XP é uma metodologia ágil para equipes pequenas e médias que desenvolvem software baseado em requisitos vagos e que se modificam rapidamente Beck (1999). Dentre as principais diferenças do XP em relação às outras metodologias estão:

- *feedback* constante;
- abordagem incremental;

- a comunicação entre as pessoas é encorajada.

Beck (1999) diz também que o XP enfatiza o desenvolvimento rápido do projeto e visa garantir a satisfação do cliente, além de favorecer o cumprimento das estimativas. As regras, práticas e valores da XP proporcionam um agradável ambiente de desenvolvimento de software para os seus seguidores, que são conduzidos por cinco valores: comunicação, simplicidade, *feedback*, coragem e respeito.

Segundo Pressman (2009) o XP prefere uma abordagem orientada a objetivos como paradigma de desenvolvimento e envolve um conjunto de regras e práticas constantes no contexto de quatro atividades metodológicas, são elas: planejamento, projeto, codificação e testes. Para aplicar os valores e princípios durante o desenvolvimento do software, o XP apresenta uma série de práticas. são elas: *Planning Game, Small Releases, Metaphor, Simple Design, Customer Tests, Sustainable Pace, Collective Ownership, Coding Standards, Test Driven Development, Refactoring e Continuous Integration*.

## 2.4 Lições do Capítulo

Neste capítulo apresentamos aspectos teóricos e tecnológicos relacionados ao desenvolvimento de jogos eletrônicos, processo de software e metodologias ágeis. Identificamos algumas lições aprendidas neste capítulo, são elas:

- os processos de software é uma sequência de atividades geralmente utilizados por uma empresa, para que os seus respectivos funcionários sigam nos seus projetos específicos;
- as metodologias ágeis foram diferente dos processos de software tradicionais, focam em valores humanos e sociais;
- o BPMN é uma técnica de modelagem gráfica de processos e é realizada através de diagramas simples, com um conjunto pequeno de elementos gráficos.

No próximo capítulo apresentamos os trabalhos relacionados que relata a execução da revisão sistemática realizada neste trabalho.



### 3 TRABALHOS RELACIONADOS

Neste capítulo reportamos o resultado de uma revisão sistemática da literatura, a qual objetivou investigar estudos relacionados à customização de processos ágeis para o desenvolvimento de jogos. Na Seção 3.1 apresentamos a definição da Revisão Sistemática e o seu objetivo e sua estrutura. Na Seção 3.2 apresentamos os resultados referentes a análise dos dados. Na Seção 3.3 apresentamos o as respostas para as questões de pesquisas definidas no protocolo da Revisão Sistemática. Por fim, na Seção 3.4, reportamos as lições aprendidas neste capítulo.

#### 3.1 Metodologia da Revisão Sistemática

O objetivo dessa revisão foi investigar estudos que propõem customizações para processos ágeis de forma a suportar o desenvolvimento de jogos. Esta revisão sistemática foi elaborada com base no guia de Kitchenham e Charters (2007). Como primeiro passo, buscamos na literatura por outras revisões sistemáticas que pudessem atender a esse escopo. Encontramos algumas revisões na literatura científica. A que mais se aproximou ao nosso objetivo foi a de O'Hagan, Coleman e O'Connor (2014), a qual teve o objetivo de encontrar processos de software utilizados no desenvolvimento de jogos. O problema desta revisão é que os autores acabam não detalhando a estrutura de cada processo encontrado.

##### 3.1.1 Questões de Pesquisa

O objetivo do estudo ficou definido como identificar as abordagens mais utilizadas nas publicações e desafios da área da utilização de metodologias ágeis no desenvolvimento de jogos do ponto de vista do desenvolvedor. Para identificar as informações necessárias neste estudo, definimos as questões de pesquisa. São elas:

1. Quais são as metodologias ágeis utilizadas para desenvolver jogos e qual é a mais utilizada?
2. Por que foram customizadas as metodologias ágeis para o desenvolvimento de jogos?
3. Que papéis, artefatos e atividades foram adaptados nas metodologias ágeis?

##### 3.1.2 Processo de Busca

A base escolhida para realizar o nosso estudo foi a biblioteca digital Scopus<sup>1</sup>. O motivo da escolha é a indexação de artigos de Computação de inúmeras origens, popularidade na academia, possui um catálogo grande e bons mecanismos de busca.

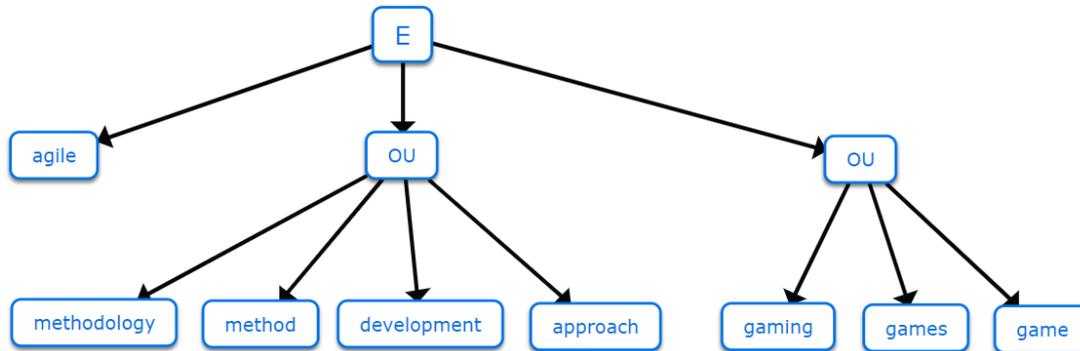
A *string* de busca é fundamental para realizar a pesquisa dos artigos referentes ao tema nas respectivas bases de dados definidas, que no caso do nosso trabalho foi a Scopus.

---

<sup>1</sup> <https://www.scopus.com>

A *string* elaborada é composta de vários termos chaves referentes ao tema e suas variações de escrita. Como visto na Figura 1, a *string* foi elaborada com três termos bases e cada termo tem variações e sinônimos.

Figura 1 – Estrutura da *string* de busca.



Fonte: Autor.

### 3.1.3 Processo de Seleção

No nosso trabalho definimos dois critérios de inclusão. Como regra, todos artigos tem que cumprir os dois critérios para ser incluído no trabalho. Os critérios de inclusão aplicados diretamente no site da Scopus e nas suas próprias opções de filtragem são:

- artigos em língua inglesa;
- não são livros.

Os critérios de exclusão definidos nesse trabalho tem como objetivo filtrar os artigos que realmente sejam importantes para o escopo do estudo. Os critérios definidos foram:

- artigos que não citem uma metodologia ágil;
- artigos que não são relacionados ao domínio de desenvolvimento de jogos;
- estudos duplicados/incompletos e indisponíveis para download;
- artigos cujo conteúdo está fora do contexto das questões de pesquisa;
- artigo que não é relacionado a metodologias de desenvolvimento ágil para jogos;
- artigos que não citem a utilização de um processo ágil para desenvolvimento de jogos.

Para aplicarmos os critérios de exclusão, analisamos título, resumo e palavras chaves de todos os artigos para identificar se algum desses critérios não foram respeitados. Se identificado que um critério não fosse respeitado, o artigo era excluído da revisão.

Os critérios de qualidade que nós definimos cobrem quatro questões:

- O artigo cita um método de desenvolvimento ágil?
- O artigo cita um processo de desenvolvimento para jogos?
- O artigo utiliza uma metodologia ágil?
- O artigo evidencia algum tipo de experimento?

Cada questão possui três opções de respostas. A média final das notas das quatro perguntas, foi considerada a nota final do artigo. As respostas podem ser:

- sim = 1.0 ponto, ou seja, a questão é respondida pelo trabalho de forma positiva;
- parcialmente = 0,5 pontos, ou seja, a questão é respondida em parte pelo artigo.
- não = 0 pontos, ou seja, esse trabalho não responde essa questão.

Os critérios de seleção foram aplicados da seguinte forma:

1. *aplicar os critérios de inclusão*- foi aplicada a *String* de busca na base de dados Scopus, os artigos que retornaram foram analisados e, se respeitarem os critérios de inclusão já estabelecidos, eles são selecionado para a próxima etapa.
2. *aplicar os critérios de exclusão*- depois de aplicar o critério de inclusão, os artigos foram submetidos aos critérios de exclusão no qual foram analisados através do título, resumo e palavras-chave e filtrados considerando os critérios de exclusão.
3. *aplicar os critérios de qualidade*- os artigos restantes seriam classificados com uma nota através dos critérios de qualidade.

#### 3.1.4 Coleta e Análise de Dados

Cada artigo lido serviu como base no preenchimento de uma ficha de leitura, composta pelos seguintes campos:

- Título do artigo;
- Autor(es) e data publicação;
- Biblioteca;
- Objetivo do artigo;
- Metodologia usada;
- Foram customizadas a(s) metodologia(s)(se houver);
- Resumo das customizações(se houver);

- Motivação para customizar (se houver);
- Papéis, artefatos e atividades customizadas (se houver);
- Resumo do artigo.

Para separar os dados interessantes para a resolução da questão de pesquisa *QP1*, foi realizada a leitura dos artigos e identificada a metodologia ágil usada em cada trabalho, com isso esses dados foram armazenados na ficha de leitura no campo Metodologia usada, e depois os dados foram agrupados e comparados através de uma tabela com a finalidade de saber quais as metodologias ágeis mais utilizadas.

Para separar os dados interessantes para a resolução da questão de pesquisa *QP2*, foi feita a leitura dos artigos e interpretado o que motivou o autor a adaptar a metodologia ágil utilizada. Com isso esses dados foram armazenados na ficha de leitura no campo Motivação para customizar, caso exista. Os dados de todos os artigos foram agrupados em uma tabela e comparados para responder a questão de pesquisa.

Já para responder a questão de pesquisa *QP3*, foi realizada a leitura dos artigos e identificada a estrutura da metodologia ágil utilizada, preenchendo na ficha de leitura o campo chamado Papéis, artefatos e atividades customizadas (se houver). Esses dados foram agrupados em uma tabela comparado os com o Scrum tradicional de Schwaber e Sutherland (2016) para sabermos o que foi alterado nessas metodologias.

## 3.2 Resultados

Nesta seção são apresentados os resultados da nossa revisão sistemática na literatura. A realização da revisão sistemática começou no dia 20/09/2018 e terminou no dia 17/10/2018, a busca direta na Scopus foi realizada no dia 05/10/2018 e obteve o retorno de 380 artigos. Após aplicarmos o critério de inclusão restaram 180 artigos, logo após o critério de exclusão restaram 12 artigos, como identificamos que possuíamos uma quantidade de trabalhos consideravelmente baixa, optamos em não aplicar o critério de qualidade, pois isso impactaria em mais trabalhos sendo excluídos do estudo. Os 12 artigos que restaram na revisão, atendem ao escopo da pesquisa e são resumidos abaixo.

Torres-Carrión et al. (2018) desenvolveram um jogo para crianças que sofrem de uma doença chamada discalculia, que é um transtorno de aprendizagem caracterizada por uma inabilidade ou incapacidade de pensar, refletir, avaliar ou raciocinar processos e tarefas que envolvam números ou conceitos matemáticos. Os autores relatam que utilizaram a metodologia ágil XP para desenvolverem o jogo e citam os vários objetivos específicos que foram estabelecidos para o cumprimento do projeto. Na conclusão os autores relatam que a metodologia XP teve uma grande ajuda para o desenvolvimento do jogo. Como é uma metodologia ágil e têm versões contínuas em cada iteração, o usuário tem um recurso sobre o qual fornecer *feedback*.

Jensen e Valente (2014) desenvolveram um jogo multiplataforma para trabalhar à formação da memória e avaliação. Outro uso do Megame inclui ortografia, formação de vocabulário e avaliação de aprendizagem de modalidade. Eles relatam que utilizaram uma abordagem ágil e iterativa no seu desenvolvimento, porém não citam a sua estrutura e como foi utilizada. Com isso, concluíram que utilizando a abordagem ágil e iterativa foi possível passar por alguns ciclos de desenvolver/teste do projeto em poucas horas: usar os protótipos para explorar rapidamente várias jogabilidades e implementar em diversos sistemas operacionais e no *android*, o que aumenta assim, os tipos de usuários que jogarão.

Petrillo e Pimenta (2010) investigaram neste artigo como os princípios e práticas dos métodos ágeis vêm sendo adotados no desenvolvimento de jogos. Segundo os autores, o processo de desenvolvimento mais utilizado é o tradicional modelo em cascata e alguns desenvolvedores utilizam o Unified Process (UP). Os autores chegam à conclusão que a adoção de práticas ágeis no desenvolvimento de jogos, podem gerar resultados promissores e que as práticas ágeis incluem as melhores características da indústria de jogos, reagindo melhor às mudanças durante o projeto e lidando com as dificuldades em modelar aspectos da experiência do usuário.

Por sua vez, Oktriono e Chong (2015) explicam o método *Application Design* aplicado na metodologia ágil Scrum. Este método permite-nos construir um aplicativo de jogo de forma mais eficiente e flexível, uma vez que nos permite melhorar nosso *design* do jogo em cada iteração do processo de Scrum. Eles chegam a conclusão, através do *feedback* do usuário que os gráficos do jogo são bons, 96% afirmaram que a aplicação mecânica de jogo vale a pena jogar, 81% concordam que a história do jogo ensina-lhes uma lição de moral e que o Scrum auxilia muito no desenvolvimento do jogo e que facilita a iteração da equipe de desenvolvimento.

Peres et al. (2011) buscaram compreender e propor ajustes para um processo de desenvolvimento de um jogo para celular. É descrito o relato de experiência do desenvolvimento do jogo, sua estrutura e processos utilizados no desenvolvimento, e ainda apresentam as reflexões sobre as adaptações necessárias para este contexto. Por fim os autores concluem que a experiência foi fundamental para aprender na prática vários conceitos da Engenharia de Software, que podem ser usados para desenvolver aplicações móveis e sociais, e que o Scrum é bem adequado para o desenvolvimento dos jogos, uma vez que o apoio a gestão da mudança, iteração, colaboração necessária em um processo de criação seja intensa e que o processo convencional tem que ser adaptado para atender os objetivos.

Mitre-Hernández et al. (2016) apresentam uma proposta de uma abordagem com o objetivo de reduzir o problema de retrabalho no desenvolvimento de jogos que geralmente começam na fase de pré-produção. Os autores mostram as adaptações necessárias na abordagem proposta e as ferramentas utilizadas no trabalho comparando a abordagem proposta por eles com a abordagem Game Design Document (GDD) do Taylor e o

desenvolvimento de jogo ágil com o Scrum. Eles concluem evidenciando o resultado do estudo e relatando diversos benefícios do Improved Game Design Document (IGDD) para reduzir o retrabalho, dizem que a abordagem proposta não só reduz o retrabalho, mas também podem dar melhores produtos de qualidade e experiência com o usuário.

Bifulco et al. (2011) descrevem desenvolvimento ágil de um jogo 3D que tem como objetivo apresentar os alunos do ensino primário os principais conceitos relacionados à coleta de resíduos e reciclagem de lixo. Segundo os autores por causa do alto nível de incerteza do projeto, foi adotado o processo de desenvolvimento baseado em Scrum e a principal característica do projeto foi a adoção de várias ferramentas de código aberto para criar e gerir o jogo e o alto nível de incerteza. Os autores concluem que o artigo descreve como um processo de desenvolvimento com base no Scrum foi aplicado para criação de um jogo sério e que alguns problemas foram encontrados durante o desenvolvimento de um jogo.

Al-Azawi, Ayesh e Obaidy (2014) relatam que o objetivo do estudo é fornecer um método de desenvolvimento de jogos, que é baseado em modelos de desenvolvimento preditivos e adaptativos. Segundo os autores a escolha pelos métodos ágeis foi influenciada por ser um modelo de desenvolvimento adaptável e diz que o modelo proposto por ele, o Agent Agile Game Development Methodology (AAGDM) é um híbrido entre o modelo preditivo utilizando a metodologia Agent Oriented Software Engineering (AOSE) e modelo adaptativo utilizando metodologia ágil. Os autores contam os problemas enfrentados na parte do planejamento do desenvolvimento do jogo e suas respectivas soluções, mostrando ferramentas utilizadas. Por fim, os autores concluem que no artigo abordaram as tendências atuais de metodologias de engenharia de software e descobriram que as vezes uma combinação desses modelos acarreta em uma melhor escolha.

Politowski et al. (2016) apresentam um levantamento dos processos de engenharia de software na indústria de jogos de vídeo a partir de análises de projeto *postmortem*. Os autores fizeram esse estudo analisando 20 autópsias do Gamasutra portal e foram extraídos os processos dos respectivos jogos e modelados através da utilização do BPMN. No final do estudo chegaram à conclusão que, o desenvolvimento de software tradicional compartilha processos e práticas semelhantes, que a indústria dos videogames tem melhorado seus processos adotando técnicas de engenharia de software, relatando assim que as práticas iterativas estão sendo utilizada mais vezes do que as demais.

Al-azawi e Ayesh (2015) apresentam um estudo de caso de como a metodologia de desenvolvimento Agent Oriented Agile Based (AOAB) foi implementada no módulo de computação móvel para criar um jogo. Os autores apresentam um projeto de jogo para uma turma, onde os alunos devem escolher entre duas metodologias de desenvolvimento, Scrum ou o AOAB, o AOAB é uma metodologia híbrida que utiliza conceitos do Scrum para sua estrutura ágil e conceitos do AOSE. Eles concluem que foi muito positiva a adoção da metodologia AOAB, na criação de um jogo em paralelo com uma metodologia

ágil, relatando assim, que isso proporcionou a obtenção de uma comparação e acrescentam que utilização de uma metodologia diferente da utilizada convencionalmente teve um retorno muito positivo.

Smith, Cooper e Longstreet (2011) evidenciam a experiência de uma turma de curso superior no desenvolvimento de um jogo, utilizando uma metodologia ágil Scrum adaptada com Cascata. Os autores relatam como foi estruturada a metodologia no cenário de sala de aula e as ferramentas utilizadas, e evidenciam algumas observações e discussões encontradas na utilização da metodologia na sala de aula. Os autores concluem que, baseado nas informações coletadas, a metodologia ágil tem ênfase no time de trabalho e lançamentos rápidos, e é mais efetivo em ajudar os alunos a produzir jogos de funcionamento. Nela relatam que forneceram aos alunos oportunidades para praticar habilidades que são altamente desejáveis para os empregados na indústria.

Asuncion et al. (2011) relatam a experiência dos alunos de graduação da Universidade de Washington no curso de Computação e Sistemas de Software, na utilização do Scrum no desenvolvimento de três jogos sérios distintos durante o ano letivo. Os autores relatam que adaptaram o cronograma de desenvolvimento limitando-os em 10 semanas para cada jogo e que um processo de Scrum com tiros de comprimento variável foi adaptado em resposta ao calendário apertado. Os autores concluem que, a equipe de desenvolvimento teve uma experiência positiva no desenvolvimento dos jogos utilizando o Scrum, mesmo com um calendário apertado e falta de experiência conseguiram entregar produtos de qualidade.

### 3.3 Respostas para as Questões de Pesquisa

A última etapa da aplicação da revisão sistemática é responder as questões de pesquisa com base nos dados analisados dos 12 artigos, considerados importantes para o escopo desse trabalho.

#### 3.3.1 Quais são as metodologias ágeis utilizadas para desenvolver jogos e qual é a mais utilizada?

Para responder essa questão nós elaboramos a Tabela 1 que exhibe os trabalhos e as metodologias ágeis citadas por cada um deles, para responder à pergunta de qual a metodologia ágil mais utilizada.

Com isso nós podemos afirmar que com base nos trabalhos relacionados provindo da revisão sistemática os processos evidenciados foram o XP citado 2 vezes, o Scrum citado 4 vezes, o Scrum Adaptado citado 2 vezes, o Scrum Híbrido citado 2 vezes e outros processos citados 4 vezes. Podemos concluir que a metodologia ágil mais utilizada foi o Scrum que foi citada 9 vezes nos 12 trabalhos, mesmo sendo adaptado ou sendo híbrido, a estrutura base do *framework* se manteve a mesma por isso foram somadas todas as

citações.

O trabalho de Oktriono e Chong (2015) é contado duas vezes na tabela 1 pelo fato do trabalho deles apresentarem dois processos, um adaptado e outro tradicional.

Tabela 1 – Metodologias ágeis utilizadas nos trabalhos relacionados.

Trabalhos	XP	Scrum	Scrum Adaptado <sup>1</sup>	Scrum Híbrido <sup>2</sup>	Outros
Torres-Carrión et al. (2018)	x	-	-	-	-
Jensen e Valente (2014)	-	-	-	-	x
Petrillo e Pimenta (2010)	x	x	-	-	-
Oktriono e Chong (2015)	-	x	x	-	-
Peres et al. (2011)	-	-	x	-	-
Mitre-Hernández et al. (2016)	-	-	-	x	-
Bifulco et al. (2011)	-	-	-	x	-
Al-Azawi, Ayeshe e Obaidy (2014)	-	-	-	-	x
Politowski et al. (2016)	-	-	-	-	x
Al-azawi e Ayeshe (2015)	-	-	-	-	x
Smith, Cooper e Longstreet (2011)	-	x	-	-	-
Asuncion et al. (2011)	-	x	-	-	-

1 Scrum modificado através de customizações em seus papéis, artefatos e eventos;

2 Scrum modificado através da junção do Scrum com outros processos.

Fonte: Autor.

### 3.3.2 Por que foram customizadas as metodologias ágeis para o desenvolvimento de jogos?

Para respondermos essa questão de pesquisa, nós interpretamos e agrupamos todos os motivos que cada autor relatou para adaptar ou utilizar uma metodologia ágil híbrida na Tabela 2. Com isso podemos visualizar que os motivos **tempo** e **facilitar as mudanças de requisitos** foram os motivos mais citados, seguidos de **adaptar ao escopo de projeto** e **diminuir o retrabalho**.

Tabela 2 – Motivos para adaptar o Scrum relatados em cada trabalho.

Trabalhos	Motivos para Adaptar			
	Adaptar ao escopo do projeto	Tempo	Diminuir retrabalho	Facilitar as mudanças de requisitos
Peres et al. (2011)	x	-	-	-
Mitre-Hernández et al. (2016)	-	-	x	-
Al-azawi e Ayeshe (2015)	-	-	-	x
Smith, Cooper e Longstreet (2011)	-	x	-	-
Al-Azawi, Ayeshe e Obaidy (2014)	-	-	-	x
Asuncion et al. (2011)	-	x	-	-

Fonte: Autor.

Al-Azawi, Ayeshe e Obaidy (2014) dizem que seu processo não é considerado o Scrum, porem utilizam como inspiração de entrada o Scrum tradicional, com isso o processo se enquadra no Scrum Adaptado.

Já Smith, Cooper e Longstreet (2011), Al-Azawi, Ayeshe e Obaidy (2014), Asuncion et al. (2011) dizem que utilizam o Scrum tradicional, porém eles alteram algumas atividades do processo, por isso se enquadra aos adaptados. Os demais trabalhos que falam que adaptaram o Scrum de alguma forma não evidenciam e justificam suas adaptações.

### 3.3.3 Que papéis, artefatos e atividades foram adaptados nas metodologias ágeis?

Como pode-se observar na Tabela 3 os papéis, atividades, artefatos que possuem o sinal (+) no começo são adaptações incluídas no processo tradicional do Scrum e os que tiverem o sinal (-) são os papéis, atividades, artefatos que foram retirados do Scrum tradicional.

Tabela 3 – Papéis, atividades e artefatos adaptados do Scrum Tradicional.

Trabalhos	Papéis	Atividades	Artefatos
Oktriono e Chong (2015)	+ Project Manager.	+ Reuniões semanais.	+ Recolha de dados; + Análise de dados; + Edifício aplicação; + Teste de aplicação.
Peres et al. (2011)	+ Stakeholders; + Project Manager.	+ Troca de informações nas quintas-feiras; + Melhorias; + Lançamento	+ Processo de documento; + Gerenciamento de configuração; + Documento de design de jogo; + Documento de arquitetura; + Atas de reuniões; + Relatório de status; + Orientações gerais; + Lista de impedimentos; + Código fonte e comentários; + Relatório de Erros; + Imagem, animações, sons e lançamento do jogo.
Asuncion et al. (2011)	- Scrum Master; - Product Owner.	+ Reunião Semanal; - Reunião diária; - Planejamento de Sprint; - Retrospectiva de Sprint; - Revisão do Sprint.	- Definição de Pronto; - Incremento do Produto; - Sprint Backlog.
Bifulco et al. (2011)		+ Programação em Pares.	

Fonte: Autor.

Os trabalhos Mitre-Hernández et al. (2016), Al-azawi e Ayeshe (2015), Smith, Cooper e Longstreet (2011), que relataram que adaptaram o Scrum, não foram citados na tabela pelo fato de não apresentarem os papéis, artefatos e atividades adaptados do Scrum tradicional.

### 3.4 Lições do Capítulo

Neste capítulo apresentamos o protocolo de revisão sistemática com seus respectivos processos, juntamente com a coleta e análise dos dados referentes a pesquisa. Identificamos algumas lições aprendidas neste capítulo, são elas:

- a revisão sistemática tem como objetivo de investigar estudos que propõem customizações para processos ágeis de forma a suportar o desenvolvimento de jogos;
- a coleta e análise dos dados foi realizada com um suporte de uma ficha de leitura;
- os resultados da revisão sistemática dão suporte aos requisitos que o processo AgileGD contem na sua elaboração.

No próximo capítulo apresentamos o processo AgileGD que possui características específicas para o desenvolvimento de jogos e segue uma lista de requisitos identificadas através da revisão sistemática presente neste capítulo.

## 4 PROPOSTA DE PROCESSO ÁGIL PARA JOGOS

Neste capítulo apresentamos o processo proposto chamado de AgileGD e como realizamos a estratégia de avaliação deste processo. Na Seção 4.1 apresentamos a proposta do processo evidenciando sua estrutura de forma detalhada. Por fim na Seção 4.3 apresentamos as lições aprendidas deste capítulo.

### 4.1 Visão Geral do AgileGD

Ao analisar os trabalhos relacionados, percebemos que muitos adaptaram as metodologias ágeis de várias formas, mas com os objetivos semelhantes. Assim, surgiu a oportunidade de unirmos as adaptações similares de cada processo relatado, a fim de avaliarmos a viabilidade do seu uso e propormos novas adaptações que se encaixem melhor para o escopo de desenvolvimento de jogos. Esse novo processo foi inspirado no Scrum e no XP e se chama AgileGD que é uma sigla derivada da palavra *Agile* e das iniciais de *Gaming Development*. O diferencial do nosso processo é evidenciado nos requisitos que ele deve conter, sendo estes:

- ser customizável - o processo tem que ser customizável a diferentes tamanhos de equipes e diferentes experiências de cada integrante, para isso utilizaremos as diretrizes de adaptação.
- possuir um guia de customização - o guia de customização tem que ser claro e objetivo para facilitar a compreensão de todos os tipos de usuários;
- respeitar jargão dos desenvolvedores - tendo em vista que nem todos da equipe serão familiarizados com os jargões da área da computação, o guia tem que ser claro para todos;
- ser voltado aos desenvolvedores de jogos - o processo tem que ser pensado para a equipe de desenvolvedores de jogos, dos mais variados tamanhos e experiências;
- respeitar os princípios ágeis - o processo tem que ser elaborado respeitando os quatro pilares do manifesto ágil.

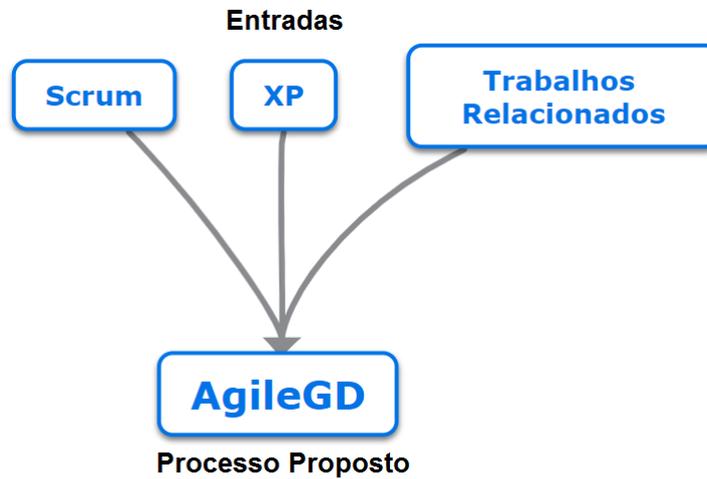
Schwaber e Sutherland (2016) dizem na sua conclusão que:

Papéis, artefatos, eventos e regras do Scrum são imutáveis e embora seja possível implementar somente partes do Scrum, o resultado não é Scrum. Scrum existe somente na sua totalidade, funcionando bem como um *container* para outras técnicas, metodologias e práticas.

Por tanto, nosso processo não é Scrum e sim inspirado no Scrum, pelo fato de termos alterado artefatos e eventos e regras.

A estrutura do processo AgileGD é inspirado em três entradas, como se pode visualizar na Figura 2. Essas entradas inspiram o processo com seus conceitos, práticas, artefatos, papéis, etc.

Figura 2 – Estrutura do AgileGD.

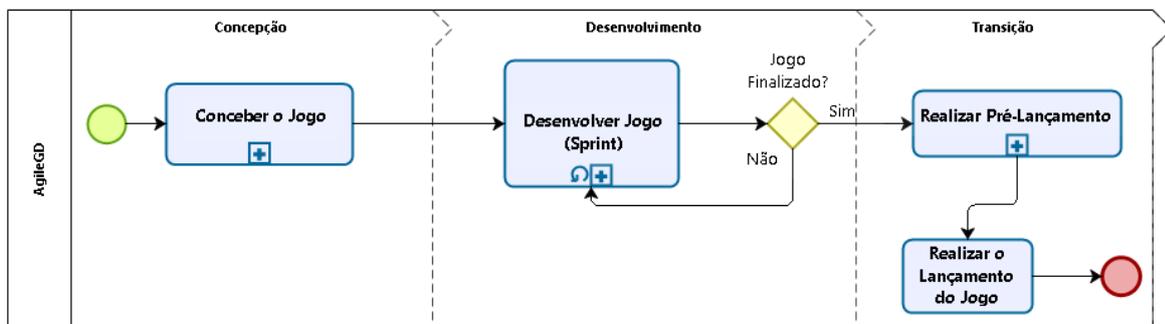


Fonte: Autor.

## 4.2 Especificação do AgileGD

O processo AgileGD é dividido em três fases principais, visíveis na Figura 3: Concepção, Desenvolvimento e Transição. Os nomes das fases foram inspirados no processo Rational Unified Process (RUP). Durante a Concepção são coletados os dados necessários para realizar o projeto. Na sequência durante a fase de Desenvolvimento são realizados todo o desenvolvimento e teste do jogo com o público interno. Por fim, durante a Transição, são realizados os últimos testes com o público externo, validações e lançamento do jogo.

Figura 3 – Processo AgileGD completo em alto nível.



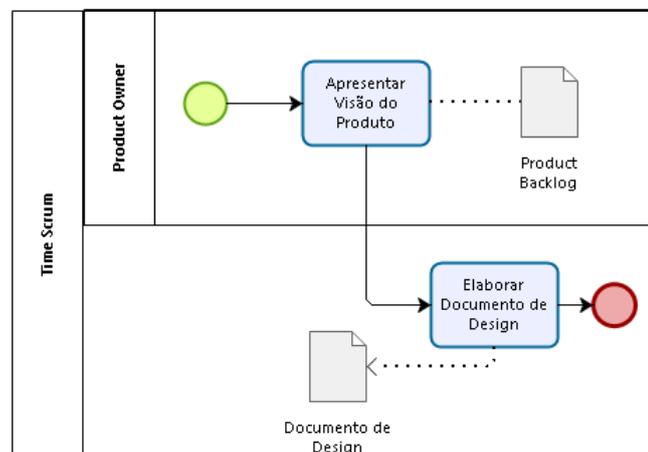
Fonte: Autor.

### 4.2.1 Fase de Concepção

A primeira fase do processo é chamada de Concepção e é um subprocesso, que é composta de apenas uma atividade, atividade essa no qual é chamada de Conceber o Jogo e é responsável por levantar os dados necessários e documenta os mesmos de forma que deem o suporte necessário para o desenvolvimento do jogo.

A atividade **Conceber Jogo** tem como objetivo levantar os requisitos necessários e organizar a arquitetura do projeto, podendo ser visualizada melhor através da Figura 4. Esse subprocesso possui duas atividades internas, são elas: Apresentar Visão do Produto e Elaborar Documento de *Design*.

Figura 4 – Subprocesso Conceber Jogo e suas atividades internas.



Fonte: Autor.

A atividade **Apresentar Visão do Produto** é o objetivo maior a ser alcançado por meio do desenvolvimento do produto, fundamental para que o projeto seja conduzido no rumo certo, como saída dessa atividade temos o *Product Backlog* que é um documento que contém a lista dos requisitos que o processo deve seguir. Essa atividade é realizada pelo *Product Owner* que é um papel do processo que representa o cliente.

Lembrando que o papel do **Scrum Master** faz parte do processo, através das figuras exibidas no processo ele não é evidenciado, porém esse papel tem como responsabilidade garantir que os valores e práticas do AgileGD estejam sendo seguidos dentro do dia a dia da equipe.

A atividade **Elaborar Documento de *Design*** representa a realização e preenchimento de um documento que contenha informações referentes ao *design* do jogo, podendo ser um *Game Design Document*, entre outros tipos de documentos. A saída dessa atividade é o artefato do Documento de *Design*, no qual é responsável por conter informações de *design* do jogo, contudo, o mesmo não possui um *template* fixo, para que o mesmo possa ser adaptável aos diferentes tipos de projetos e equipes, mas o documento deve possuir informações referentes a algumas áreas consideradas necessárias para

a maioria dos projetos, são elas: conceito, especificações técnicas, especificações do jogo, dispositivos de entrada, *design* gráfico e arte, sonorização e desenvolvimento. Essa atividade é realizada pelo Time de Desenvolvimento que é composto pelos desenvolvedores e especialistas pertencentes ao projeto.

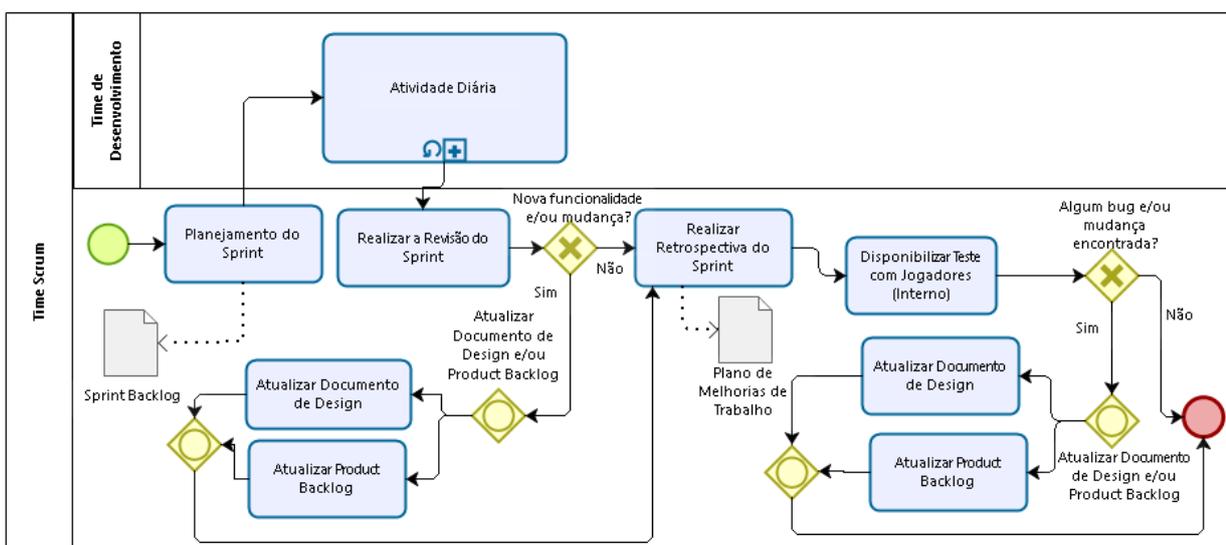
#### 4.2.2 Fase de Desenvolvimento

A fase de Desenvolvimento é o coração do processo AgileGD e possui todo ciclo de desenvolvimento do jogo. Esse ciclo chama-se *Sprint* e é evidenciado no subprocesso Realizar Desenvolvimento do Jogo (*Sprint*) apresentada na Figura 3.

Esse subprocesso é denominada **Realizar Desenvolvimento do Jogo (*Sprint*)** e apresenta o ciclo de desenvolvimento do jogo chamado de *Sprint*. Nela se realizam as atividades Realizar Planejamento do *Sprint*, Atividade Diária, Realizar Revisão do *Sprint*, Atualizar *Product Backlog* e Realizar a Retrospectiva do *Sprint*, Disponibilizar Teste com Jogadores (Interno) e Corrigir *Bug* ou Mudança apresentados na Figura 5.

O evento ***Sprint*** não é visível na Figura 3, porém ele representa todo ciclo interno da atividade Realizar Desenvolvimento do Jogo. O *Sprint* retrata um período de tempo em que são desenvolvidos os requisitos do jogo. Esse período tem uma duração estipulada por cada equipe do projeto, porém vale ressaltar que *Sprints* muito longos tendem a ter escopos maiores. A ideia aqui é que o desenvolvedor tenha seu objetivo bem delimitado, e mantenha seu foco nas atividades daquele determinado *Sprint*, para que no fim dele já sejam testadas as funcionalidades desenvolvidas.

Figura 5 – Subprocesso Realizar Desenvolvimento do Jogo (*Sprint*) e suas atividades internas.



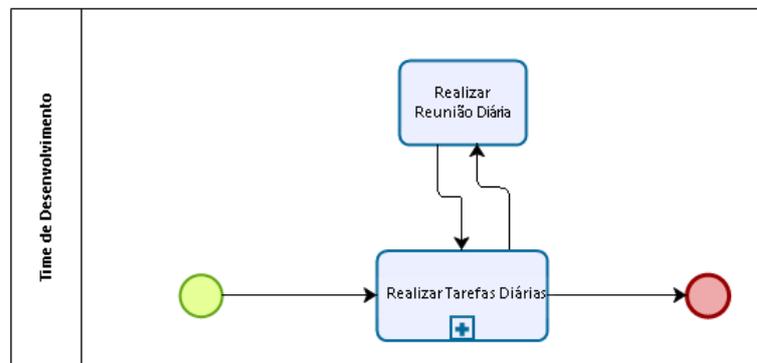
Fonte: Autor.

A atividade **Planejamento do *Sprint*** se define como uma reunião com todos

integrantes da equipe, no caso o Time Scrum, com o objetivo de definirem quais requisitos serão elaborados no *sprint* que está a iniciar. Essa atividade tem como saída o artefato *Sprint Backlog*, que é um documento contendo os requisitos daquele *Sprint* específico. Essa atividade é realizada pelo Time Scrum, que é composto pelo *Product Owner*, *Scrum Master* e pelo Time de Desenvolvimento.

O subprocesso **Atividade Diária** apresenta o ciclo que é realizado durante os dias pertencentes a um *Sprint*, suas atividades internas são: Realizar Tarefas Diárias e Realizar Reunião Diária. Seu fluxo de execução pode ser visualizada na Figura 6. Essa atividade é realizada pelo Time de Desenvolvimento.

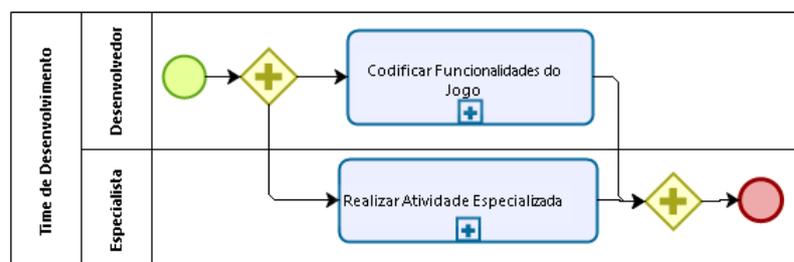
Figura 6 – Subprocesso Atividade Diária e suas atividades internas.



Fonte: Autor.

O subprocesso **Realizar Tarefas Diárias** apresenta o ciclo que a ser feito para desenvolver, testar e modelar o jogo, suas atividades são: Codificar Funcionalidades do Jogo e Realizar Reunião Diária e pode ser melhor visualizado através da Figura 6. Essa atividade também é realizada pelo Time de Desenvolvimento.

Figura 7 – Subprocesso Realizar Tarefas Diárias e suas atividades internas



Fonte: Autor.

O subprocesso **Codificar Funcionalidades do Jogo** apresenta as fases que serão feitas em paralelo diariamente pelos desenvolvedores, são elas: Analisar, Projetar, Codificar e Testar e pode ser visualizada na Figura 8. Essa atividade é realizada pelo

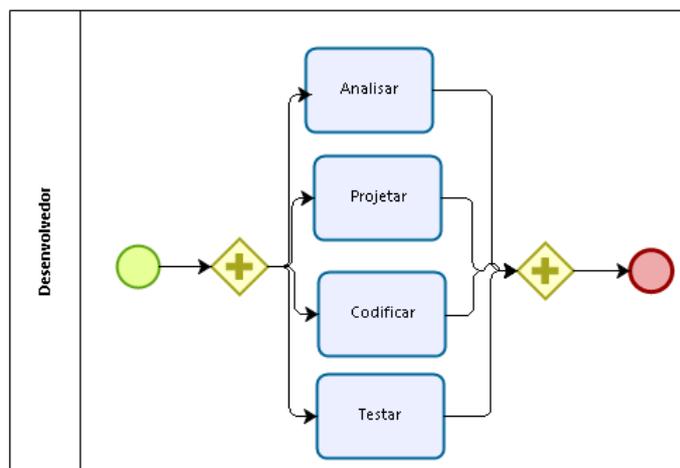
Desenvolvedor. Algumas práticas inspiradas no XP vão ser utilizadas em paralelo com essas atividades e outra pode ser utilizada, essas práticas são apresentadas abaixo.

A prática **Design Simples**, é realizada em conjunto com a atividade Codificar Funcionalidades do Jogo (*Sprint*), pois é uma atividade muito importante, seu objetivo é sempre priorizar o código do jogo que está sendo desenvolvido na forma mais simples e mais clara possível, conforme os padrões definidos pela equipe de desenvolvimento, facilitando a compreensão e possível continuidade por qualquer um de seus membros.

A prática **Padronizar**, é realizada em conjunto com a atividade Codificar Funcionalidades do Jogo, pois é uma atividade muito importante, tendo em vista que seu objetivo é de que todo código desenvolvido deve ter um padrão e todos devem seguir o mesmo padrão. Assim, se tem a impressão de que o código-fonte foi implementado e editado pela mesma pessoa, independente da quantidade de integrantes, facilitando a elaboração, manutenção e atualizações do jogo.

A prática **TDD** pode ser utilizado em conjunto com a atividade Realizar Desenvolvimento do Jogo, tendo em vista que ela é uma atividade opcional, ou seja, podendo ser ou não utilizada pela equipe de desenvolvimento, seu objetivo é que os testes sejam escritos antes da produção do código, onde para cada funcionalidade do sistema um teste é criado anteriormente.

Figura 8 – Subprocesso Codificar Funcionalidades do Jogo.



Fonte: Autor.

O subprocesso **Realizar Atividade Especializada** apresenta as atividades que são feitas em paralelo diariamente pelos especialistas, que podem ser *designer*, artista, *designer* de som, entre outros papéis que a equipe possui. Essa atividade é apresentada como um subprocesso, porém não são evidenciados as suas atividades internas, deixando-a flexível e podendo ter atividades internas que se encaixem a cada equipe, como: Realizar *Designer* de Personagem, Realizar Efeito de Gráfico do Cenário, Projetar o Cenário, Gravar os Efeitos Sonoros, entre outros. Para auxiliar na adaptação dessa atividade no

Apêndice B é apresentado um Guia de Adaptação contendo informações referentes a essa atividade. Atividade que é realizada pelo Especialista.

A atividade **Realizar Reunião Diária** é realizada diariamente, podendo ser realizada no começo do expediente ou no fim do expediente, com o objetivo de forçar a comunicação de toda equipe, para que todos possam sincronizar suas atividades e saber o andamento das tarefas de cada membro da equipe de desenvolvimento, podendo ser visualizada na Figura 6. Essa atividade é realizada pelo Time de Desenvolvimento.

Em apoio a esse atividade é utilizada uma prática de **Reunião em Péque**, somada à reunião diária, auxilia na agilidade da reunião, tendo em vista que uma reunião sentada dá mais conforto e conseqüentemente pode ocasionar em uma reunião mais estendida.

A atividade **Realizar Revisão do *Sprint*** é uma reunião com o objetivo de analisar o incremento do produto que é a parte do jogo desenvolvida no evento *Sprint* e adaptar o *Product Backlog* se necessário. Seu objetivo não é receber aprovação e sim obter *feedback*. Essa atividade é realizada pelo Time Scrum.

A atividade **Atualizar Documento de *Design*** representa a ação que pode ser realizada se uma nova funcionalidade e/ou mudança for identificada na atividade Realizar Revisão do *Sprint*, atualizando o artefato Documento de *Design* com as respectivas mudanças necessárias. Essa atividade é realizada pelo Time Scrum.

A atividade **Atualizar *Product Backlog*** representa a ação que pode ser realizada se uma nova funcionalidade e/ou mudança for identificada na atividade Realizar Revisão do *Sprint*, atualizando o artefato *Product Backlog* com as respectivas mudanças necessária. Essa atividade é realizada pelo Time Scrum.

A atividade **Realizar Retrospectiva do *Sprint*** representa a última reunião do *Sprint*. Seu principal objetivo é rever o andamento da última *Sprint* e criar como saída um artefato chamado Plano de Melhorias de Trabalho para ser aplicada na próxima *Sprint*. Essa reunião é feita pelo Time Scrum e o *Product Owner* deve ficar de fora, a menos que seja convidado.

A atividade **Disponibilizar Teste com Jogador (Interno)** tem como o objetivo disponibilizar para a própria equipe a parte do jogo (incremento) que foi elaborada no respectivo *Sprint*, para que seja identificado *bugs* ou alguma mudança no projeto através do teste de jogabilidade. Essa atividade é realizada pelo Time Scrum.

A atividade **Atualizar Documento de *Design*** representa a ação que pode ser realizada se uma nova funcionalidade e/ou mudança for identificada na atividade Disponibilizar Teste com Jogadores (Interno), atualizando o artefato Documento de *Design* com as respectivas mudanças necessárias. Essa atividade é realizada pelo Time Scrum.

A atividade **Atualizar *Product Backlog*** representa a ação que pode ser realizada se uma nova funcionalidade e/ou mudança for identificada na atividade Disponibilizar Teste com Jogadores (Interno), atualizando o artefato *Product Backlog* com as respectivas mudanças necessária. Essa atividade é realizada pelo Time Scrum.

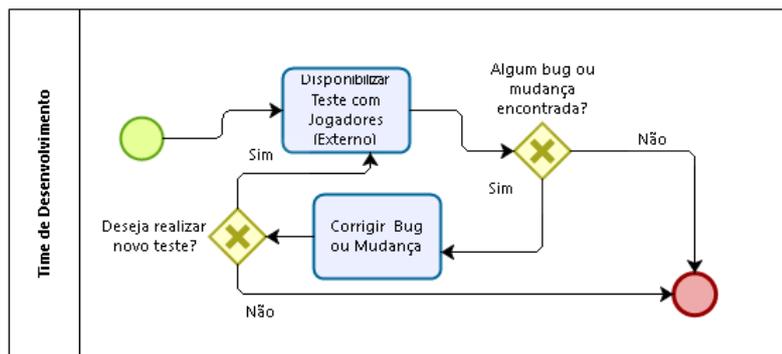
### 4.2.3 Fase de Transição

A fase de Transição apresenta as atividades de encerramento do projeto, são elas: Realizar Pré-Lançamento do Jogo e Realizar Lançamento do Jogo como pode ser visto na Figura 3.

O subprocesso **Realizar Pré-Lançamento do Jogo** é responsável por disponibilizar uma versão do jogo para o público externo, com objetivo de retornar *feedback* relacionados a *bugs* e melhorias, no caso é o último ajuste antes do lançamento do jogo. Esse subprocesso possui duas atividades internas chamadas Disponibilizar Teste do Jogo (Externo) e Corrigir Bug ou Mudança que podem ser melhor visualizadas na Figura 9.

A atividade **Disponibilizar Teste do Jogo (Externo)** é responsável por fazer os últimos testes do produto todo, com o objetivo de retornar o *feedback* de jogadores fora da equipe de desenvolvimento. O período de realização do teste será definido por cada equipe. Essa atividade é realizada pelo Time de Desenvolvimento.

Figura 9 – Subprocesso Pré-Lançamento do Jogo e suas atividades internas.



Fonte: Autor.

A atividade **Corrigir Bug ou Mudança** ocorre depois de realizar a atividade Disponibilizar Teste com Jogadores (Externo), ou seja, se um *bug* ou mudança for identificada durante esse teste a equipe lança uma atualização para corrigi-la. Essa atividade é realizada pelo Time de Desenvolvimento.

A atividade **Realizar Lançamento do Jogo** como pode ser vista na Figura 3 é responsável pela finalização do projeto, tendo em vista que o jogo já foi validado internamente e validado externamente, o mesmo está preparado para ser lançado ao público final.

## 4.3 Lições do Capítulo

Nesse capítulo apresentamos o processo AgileGD com seu fluxo de execução e sua estrutura sendo apresentada de forma didática e detalhada, para que os usuários não tenham dificuldade em entender o processo. Algumas lições aprendidas foram identificadas nesse capítulo, são elas:

- o processo AgileGD utiliza três entradas para a inspiração dos seus papéis, atividades, artefatos, práticas, entre outros;
- o processo AgileGD foi modelado graficamente utilizando o BPMN.

No próximo capítulo apresentamos a técnica de avaliação do AgileGD, apresentando seu planejamento, realização, resultados e as ameaças encontradas.



## 5 AVALIAÇÃO DO AGILEGD

Neste capítulo apresentamos a técnica para avaliar e validar o processo proposto chamada de Grupo Focal conforme proposta no artigo de Kontio, Bragge e Lehtola (2008). O principal objetivo dessa estratégia de avaliação é validar o processo e levantar novos requisitos com o auxílio das opiniões de especialistas do assunto, com isso, enriquecer ainda mais o processo proposto. Foi aplicado esse experimento com graduandos, mestrando e mestres no ambiente da Universidade Federal do Pampa. Na Seção 5.1 apresentamos o planejamento da execução do Grupo Focal. Na Seção 5.2 apresentamos a execução e análise dos resultados do Grupo Focal. Por fim na Seção 5.3 apresentamos as lições aprendidas nesse capítulo.

### 5.1 Planejamento

A motivação para a realizar o Grupo Focal é a validação é obter *feedback* relacionado ao processo proposto, a fim de ter um ponto de vista de pessoas com experiência no tema relacionado ao estudo. Os participantes do Grupo Focal são estudantes de graduação, mestrando, graduados e mestres, que tenham alguma experiência com metodologias ágeis e desenvolvimento de jogos. Como segmentação procuramos equilibrar o grupo de participantes, juntando aqueles com bastante experiência com os participantes com pouca experiência. O cenário para aplicação do Grupo Focal é a sala de reuniões localizada no Parque Científico e Tecnológico do Pampa, que é o órgão da Universidade Federal do Pampa responsável por incentivar o empreendedorismo e a inovação tecnológica.

O roteiro do Grupo Focal possui a sequência de atividades apresentada abaixo:

- preliminares - boas-vindas aos participantes, preparação dos participantes para a realização do Grupo Focal e apresentando para os mesmos, o roteiro do Grupo Focal e como ele é realizado;
- objetivo principal - apresentação do objetivo principal do debate;
- contexto - apresentação do contexto do processo, relatando os problemas relatados na literatura.
- primeira tarefa - apresentação do Termo de Consentimento Livre e Esclarecido e o Questionário de Nivelamento;
- conceitos - apresentação dos conceitos sobre Jogos Eletrônicos e Processo de Desenvolvimento;
- discussão - apresentação de um problema e a possível solução desse problema, acompanhados do Material de Apoio e após o período de debate disponibilização do questionário de participação;

- encerramento - agradecimento aos participantes pela participação.

O documento do termo Termo de Consentimento Livre e Esclarecido (TCLE), Questionário de Nivelamento, Material de Apoio e o Questionário de Participação podem ser visualizados no Apêndice A.

A captura de dados é feita através dos dados preenchidos pelos participantes no questionário de participação e gravações/anotações dos observadores sobre pontos importantes discutidos. Por fim analisaremos esses dados com o objetivo de refatorar o processo.

O papel do moderador foi definido, tendo como objetivo explicar sobre as dificuldades referentes ao processo e instigar os participantes a debater sobre os problemas apresentados. O observador foi definido como o responsável por anotar comentários dos participantes considerados relevantes para o processo, sendo eles negativos ou positivos. Ambos papéis foram realizados por estudantes da graduação em Engenharia de Software.

## 5.2 Execução

O moderador iniciou o Grupo Focal recebendo os participantes na sala de reuniões do Parque Científico e Tecnológico do Pampa no dia 29/05 às 08:40 AM. Os participantes foram compostos por 4 graduandos em Engenharia de Software, 1 estudante de Mestrado em Engenharia de Software e 1 mestre em Ciência da Computação. Após a recepção dos participantes, o moderador apresentou o roteiro do Grupo Focal e definiu um Grupo Focal e o contexto da sua execução. Na sequência foi distribuído o Termo de Consentimento Livre e Esclarecido como a primeira tarefa, e informou aos participantes que se os mesmos concordassem com o presente termo deveriam assiná-lo. Na sequência ainda na primeira tarefa, foi distribuído aos participantes o questionário de nivelamento, o qual tem como objetivo saber o nível de conhecimento sobre os conceitos relacionados ao processo proposto.

Após essas atividades primárias, foi apresentado o problema 1 sendo este: “Como incluir o documento de *design* nesse novo processo?”, em seguida foi apresentado pelo moderador uma possível solução para esse problema acompanhado da disponibilização do material de apoio aos participantes. O participante 2 relatou que “A forma que foi solucionado esse problema é muito coerente, porém, é muito necessário deixar bem evidente que o documento de *design* poderá ser atualizado”, o participante 5 por sua vez relatou que “Concordo com a fala do participante 2, e com base no material de apoio sinto falta do tópico mecânica do jogo na estrutura do documento de *design*.”. Ademais, o participante 3 disse “Ao meu ver está parcialmente ok, pois deveria existir antes do documento de *design* uma análise do mercado, talvez estar no próprio artefato uma atividade mais voltada à análise do mercado.”, após esses debates foi disponibilizado aos participantes um questionário de participação referente ao problema 1, para que os mesmos informassem

os seus respectivos pontos de vista em relação a solução apresentada.

Na sequência, logo após os participantes preencherem o questionário de participação do problema 1, o moderador apresentou o problema 2 sendo este: “Como resolver o problema de falta de suporte aos diferentes tipos e tamanhos de equipes nesse novo processo?”, após a apresentação do problema foi apresentado pelo moderador a possível solução e a disponibilização do material de apoio aos participantes. O participante 1 disse que “A repartição dos papéis é coerente ao problema, mas não sei se da forma que essa atividade foi apresentada, ela poderá ser interpretada de outra forma pelos usuários.”. Em apoio a essa afirmação o participante 6 alega que “O participante 1 tem razão, os usuários podem confundir o objetivo que é a liberdade de adaptação com anarquia.”. O participante 5 afirma “Uma forma de resolver esses problemas levantados é a atividade possuir um guia de adaptação, com um processo para instanciar esse subprocesso Realizar Atividade Especializada.”, após esses debates foi disponibilizado aos participantes um questionário de participação referente ao problema 2, para que os mesmos informassem os seus respectivos pontos de vista em relação a solução apresentada.

Logo após os participantes preencherem o questionário de participação do problema 2, foi apresentado pelo moderador o problema 3 sendo este: “Como incrementar o teste de jogabilidade ainda na fase de desenvolvimento do projeto?”, em seguida foi apresentado pelo moderador uma possível solução para esse problema acompanhado da disponibilização do material de apoio aos participantes. O participante 1 afirma que “Faz sentido o Time *Scrum* ser responsável pela atividade Disponibilizar Teste com Jogador (Interno), mas acho que deveria trocar o texto evidente na figura do processo, invés de “mudança” colocar outra palavra como por exemplo “alteração””. Já o participante 6 diz que “Discordo no quesito mudar o texto da figura, acho que da forma que tá escrito tá ok, mas eu acho que deveria ter uma condicional dando ao usuário a escolha de teste com público interno ou teste com público externo”, após esses debates foi disponibilizado aos participantes um questionário de participação referente ao problema 3, para que os mesmos informassem os seus respectivos pontos de vista em relação a solução apresentada.

Por fim, após os participantes preencherem o questionário de participação do problema 3, o moderador apresentou o problema 4 sendo este: “Como incrementar o teste de jogabilidade na fase do pós-projeto do jogo?”, logo em seguida foi apresentado pelo moderador uma possível solução para esse problema acompanhado da disponibilização do material de apoio aos participantes. O participante 4 relatou após analisar o material que “Se existir a opção do teste interno e externo no *Sprint*, não teria motivo para existir essa atividade no Pré-Lançamento do jogo”. O participante 3 disse que “Discordo que não tenha motivos para existir essa atividade no Pré-Lançamento do Jogo, mas acho que deve existir uma forma de voltar a fase de desenvolvimento (*Sprint*) se caso alguma *feature* fosse desejada”. Por sua vez, o participante 1 disse “Além de possuir a opção de realizar uma nova *feature*, deverá existir a possibilidade de atualizar o *Product Backlog*

e o Documento de *Design* mesmo no Pré-Lançamento do Jogo”, após esses debates foi disponibilizado aos participantes um questionário de participação referente ao problema 4, para que os mesmos informassem os seus respectivos pontos de vista em relação a solução apresentada. Após os participantes preencherem o questionário de participação do problema 4, o moderador agradeceu a participação de todos e finalizou o Grupo Focal às 11:20 AM.

### 5.2.1 Resultados

A partir do debate realizado no Grupo Focal envolvendo quatro problemas identificados na elaboração de um processo específico para jogos, chegamos às respostas identificadas através do questionário de participação que pode ser visualizado no Apêndice A, pontos fortes e oportunidades de melhorias identificados através do questionário de participação e dos dados coletados do debate (áudio/anotações).

#### 5.2.1.1 Problema 1

O problema 1 tem como objetivo apresentar o problema identificado durante o estudo “Como incluir o artefato documento de *design* nesse novo processo?”, com isso foram elaboradas quatro frases afirmativas, com as opções de resposta sendo: discordo fortemente, discordo, indiferente, concordo e concordo fortemente, as frases são apresentadas a baixo.

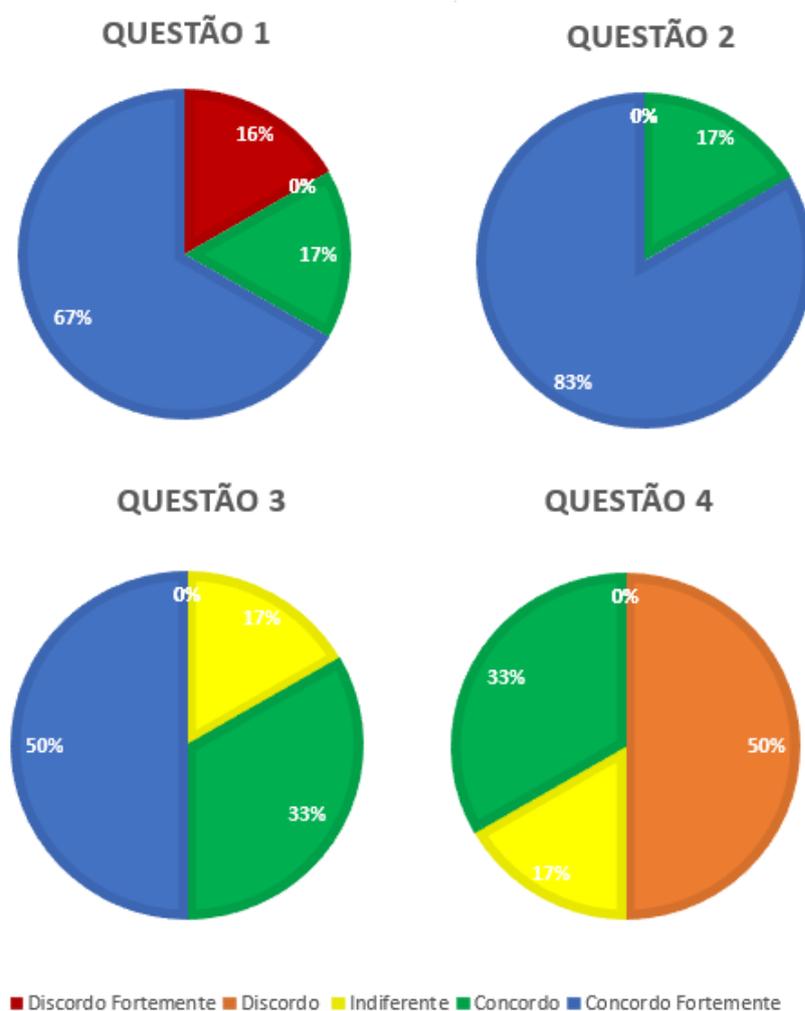
- Questão 1 - deve existir uma atividade responsável para elaborar o documento de *design* do jogo no processo;
- Questão 2 - deve existir um documento de *design* do jogo no processo;
- Questão 3 - o nome artefato ser um nome genérico (Documento de *Design*), tendo em vista que esse documento poderá ser adaptável aos diferentes tipos de equipes e possuir diferentes formatos;
- Questão 4 - a solução apresentada resolve todos os problemas identificados.

No gráfico apresentado na Figura 10 é possível visualizar os resultados referentes a cada questão do questionário de participação disponibilizados para os participantes.

Os pontos fortes relatados através do debate (áudio/anotações) e o questionário de participação sobre o problema 1 são: o documento de *design* ser adaptável aos diferentes tipos de equipes e projetos e o documento de *design* possuir um nome genérico para facilitar o entendimento dos usuários que o mesmo não possui um *template* fixo.

A oportunidade de melhoria identificada pelos participantes sobre o problema 1, foi a de deixar evidente na descrição do documento de *design* que ele poderá ser atualizado conforme o decorrer do projeto, essa oportunidade de melhoria foi incluída no processo

Figura 10 – Gráficos de Pizza referentes as respostas do problema 1.



Fonte: Autor.

com o objetivo que ainda na fase de concepção já será identificado que o documento de *design* será atualizado conforme o decorrer do projeto, facilitando o entendimento dos usuários.

#### 5.2.1.2 Problema 2

Já o problema 2 tem como objetivo apresentar o problema identificado durante o estudo “Como resolver o problema de falta de suporte aos diferentes tipos e tamanhos de equipes nesse novo processo?”, com isso foram três frases afirmativas, com as opções de resposta sendo: discordo fortemente, discordo, indiferente, concordo e concordo fortemente, as frases são apresentadas a baixo.

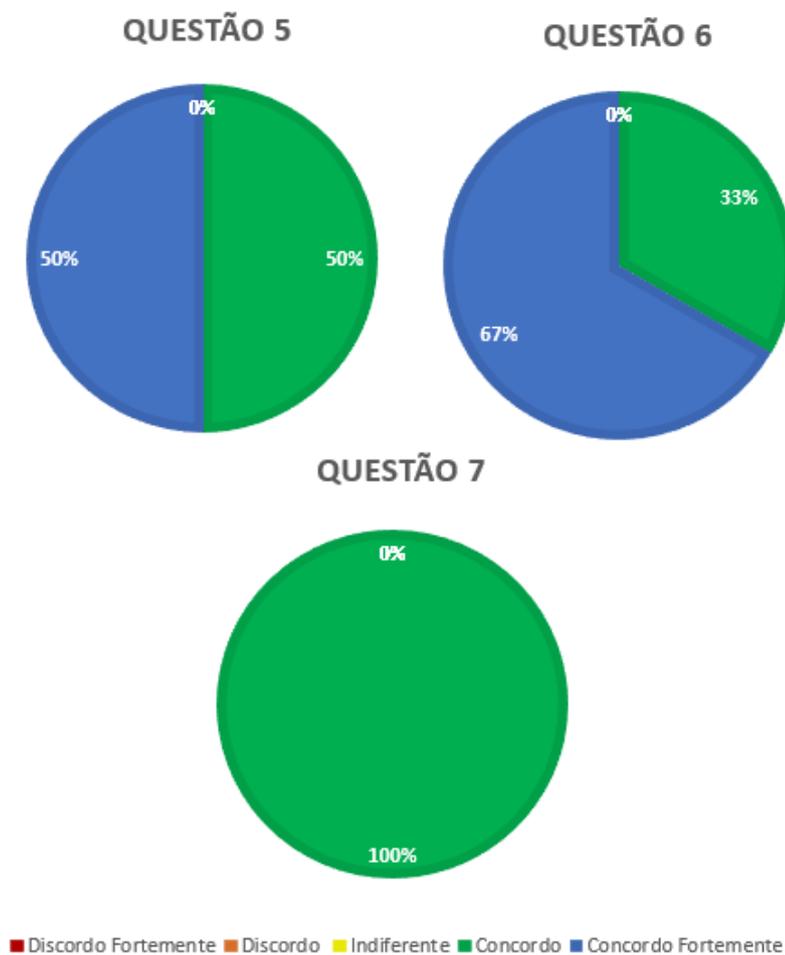
- Questão 5 - separar o time de desenvolvimento, manter um papel responsável pelo desenvolvimento do jogo (codificação) e um papel que englobe os outros papéis do

desenvolvimento do jogo (tarefas que não envolvem codificação).

- Questão 6 - o nome do subprocesso é um nome genérico (Realizar Atividades Especializadas), tendo em vista que esse subprocesso poderá ser adaptável aos diferentes tipos de equipes e possuir diferentes tipos de atividades internas.
- Questão 7 - a solução apresentada resolve todos os problemas identificados.

No gráfico apresentado na Figura 11 é possível visualizar os resultados referentes a cada questão do questionário de participação disponibilizados para os participantes.

Figura 11 – Gráficos de Pizza referentes as respostas do problema 2.



Fonte: Autor.

Os pontos fortes relatados através do questionário de participação, áudio e anotações sobre o problema 2 são: Separar a equipe em desenvolvedores e especialistas resolve na teoria o problema relatado sobre os processos não darem “espaço” aos especialistas envolvidos no projeto.

A oportunidade de melhoria identificada pelos participantes sobre o problema 2, foi que a atividade Realizar Atividade Especializada deveria possuir um processo e um

guia de adaptação para auxiliar o entendimento do usuário, essa oportunidade de melhoria foi incluída no processo assim facilitando o entendimento do futuro usuário de como essa atividade será adaptável e como ele fará para adaptar a mesma.

### 5.2.1.3 Problema 3

O problema 3 por sua vez, tem como objetivo apresentar o problema identificado durante o estudo “Como resolver o problema de falta de suporte aos diferentes tipos e tamanhos de equipes nesse novo processo?”, com isso foram elaborados três frases afirmativas, com as opções de resposta sendo: discordo fortemente, discordo, indiferente, concordo e concordo fortemente, as frases são apresentadas a baixo.

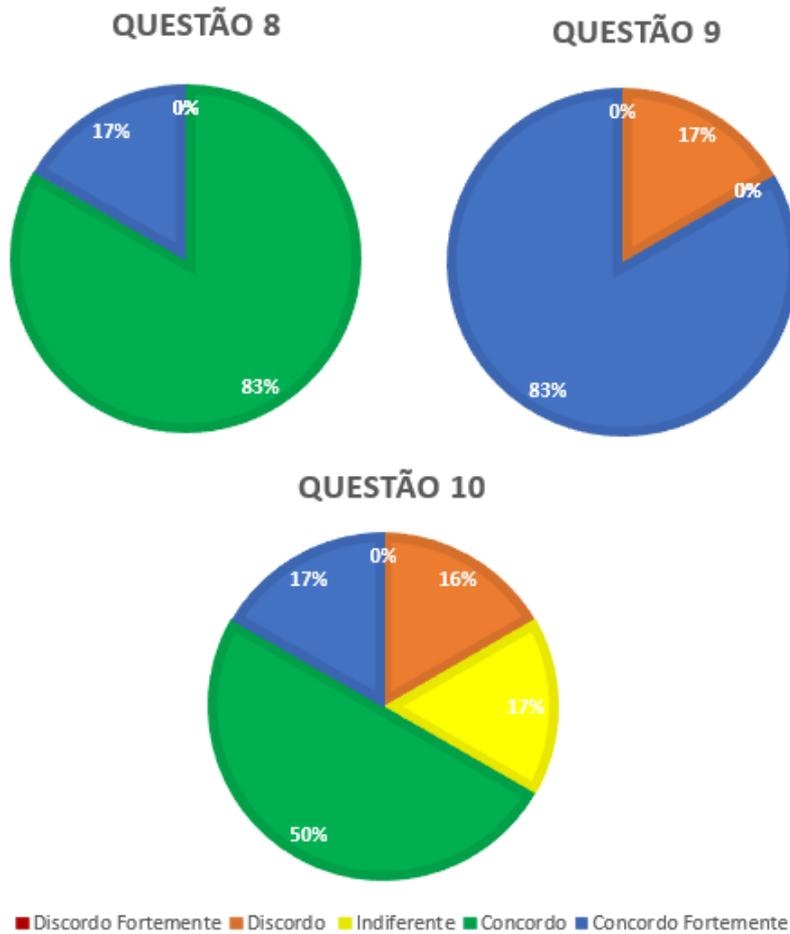
- Questão 8 - deve existir um teste de cada funcionalidade (incremento) que foi desenvolvida no *Sprint*, com o público do teste de usabilidade sendo a própria equipe de desenvolvimento.
- Questão 9 - deve-se chamar a atividade de “Realizar Teste com Jogador (Interno)” e não chamar de versão “Alfa”, “Pré Alfa” e entre outros nomes por exemplo, deixando assim para a equipe decidir as características do teste de jogabilidade da forma que melhor se encaixa ao escopo do projeto.
- Questão 10 - a solução apresentada resolve todos os problemas identificados.

No gráfico apresentado na Figura 12 é possível visualizar os resultados referentes a cada questão do questionário de participação disponibilizados para os participantes.

Os pontos fortes relatados através do questionário de participação, áudio e anotações sobre o problema 3 são: O teste de jogabilidade do incremento ser voltado ao público interno e ser no final da *Sprint* e o nome da atividade ser “Realizar Teste com Jogadores (Interno)” é adequado perante a proposta da atividade possuir um nome genérico.

Algumas oportunidades de melhorias foram identificadas pelos participantes sobre o problema 3, algumas foram incluídas no processo e outras estão em análise ou foram rejeitas, são elas: A oportunidade de melhoria que alegava que deveria existir junto com a atualização do *Product Backlog* a atualização do Documento de *Design*, foi incluída no processo porque nós identificamos que essa oportunidade de melhoria deveria ser colocada no processo, assim o Documento de *Design* poderá ser atualizado toda vez que uma mudança for identificada no processo. Já a oportunidade de melhoria que alega que deveria possuir a opção de realizar um teste com o público interno ou externo no final da *Sprint*, está em análise pelo fato de que ainda falta validação e *feedback* da indústria especializada, para identificar se realmente é realizada essa estratégia na indústria.

Figura 12 – Gráficos de Pizza referentes as respostas do problema 3.



Fonte: Autor.

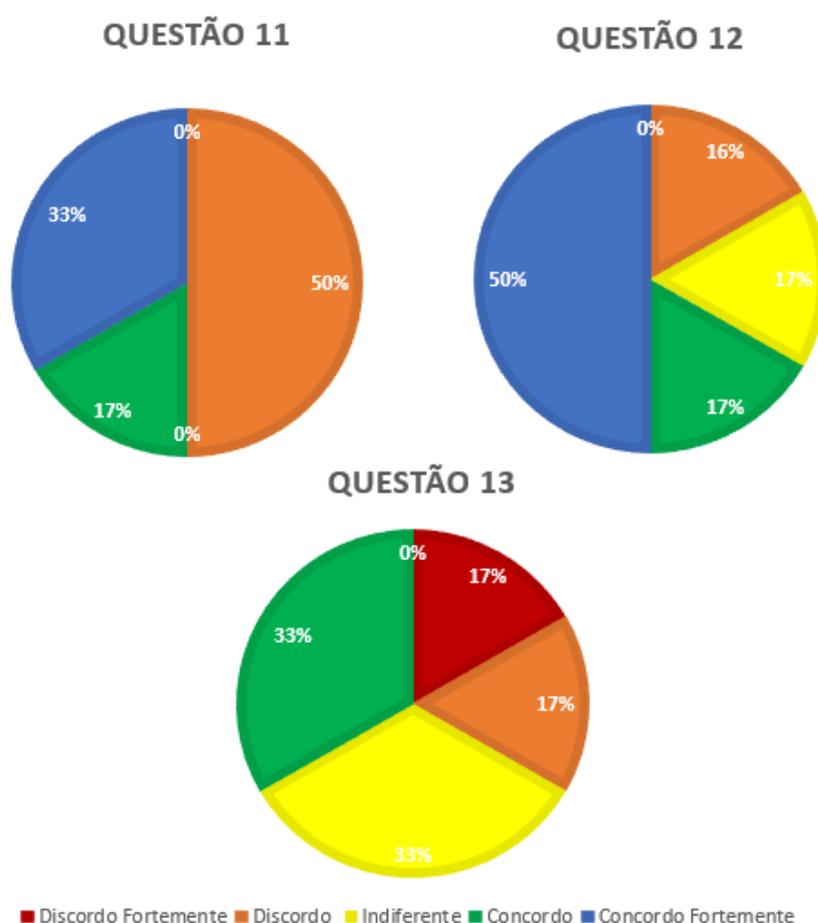
#### 5.2.1.4 Problema 4

Por fim, o problema 4 tem como objetivo apresentar o problema identificado durante o estudo “Como incrementar o teste de jogabilidade na fase do pós-projeto do jogo?”, com isso foram elaborados três frases afirmativas, com as opções de resposta sendo elas: discordo fortemente, discordo, indiferente, concordo e concordo fortemente, as frases são apresentadas a baixo.

- Questão 11 - deve existir um teste do produto final (jogo), com o público do teste jogabilidade sendo usuários fora do projeto de desenvolvimento.
- Questão 12 - deve se chamar de “Realizar Teste com Jogador (Externo)” e não chamar de versão “Beta”, “Open Beta”, “Release Candidate (RC)” e entre outras nomes, por exemplo, deixando assim para a equipe decidir as características do teste de usabilidade da forma que melhor se encaixa ao escopo do projeto.
- Questão 13 - a solução apresentada resolve todos os problemas identificados.

No gráfico apresentado na Figura 13 é possível visualizar os resultados referentes a cada questão do questionário de participação disponibilizados para os participantes.

Figura 13 – Gráficos de Pizza referentes as respostas do problema 4.



Fonte: Autor.

Os pontos fortes relatados através do questionário de participação, áudio e anotações, sobre o problema 4 são: O nome da atividade ser “Realizar Teste com Jogadores (Externo)”, se mostra adequado para o objetivo da atividade em possuir um nome genérico e a forma que foi pensado estruturalmente o teste de jogabilidade do produto final, sendo executado na fase de Pré Lançamento do jogo se mostrou correta.

Algumas oportunidades de melhorias foram identificadas pelos participantes sobre o problema 3, algumas foram incluídas no processo e outras estão em análise ou foram rejeitas, são elas: Algumas oportunidades de melhorias foram identificadas pelos participantes sobre o problema 4, são elas: A oportunidade de melhoria que questionava o nome da Fase ser “Pós-Projeto”, segundo os participantes esse nome remete ao entendimento que o projeto já foi finalizado e acaba confundindo os usuários, essa oportunidade de melhoria foi incluída no projeto e o nome da fase passou a ser “Transição”, facilitando assim

aos usuários o entendimento sobre o objetivo dessa fase. Já a oportunidade de melhoria que alega que deveria existir a possibilidade do jogo mesmo estando na atividade de Pré Lançamento, poder voltar para o ciclo de desenvolvimento (*Sprint*), a oportunidade de melhoria que relata que para novas *features* serem implementadas por exemplo, o teste de jogabilidade na Pré Lançamento do Jogo deveria ser opcional, a oportunidade de melhoria que diz deveria possuir a opção de alterar o *Product Backlog* e o Documento de *Design* mesmo estando no Pré Lançamento do jogo e a oportunidade de melhor que diz o teste de jogabilidade com o público externo poderia ser realizado durante o desenvolvimento do jogo (*Sprint*) estão em análise pois necessita-se de mais *feedback* e validação com especialistas para saber a viabilidade de incrementar no processo essas funcionalidades.

### 5.2.2 Ameaças

Uma ameaça à validade do Grupo Focal é que o mínimo de participantes planejado para o Grupo Focal deveria ser em torno de 4 integrantes, com isso foram 6 participantes efetivos, com isso houve uma certa segurança em relação a realização do Grupo Focal. Outra ameaça foi os integrantes não estarem se sentindo à vontade ou motivados para debaterem, com isso foi disponibilizado café, água, biscoitos e chocolates para que os mesmo se sintam mais à vontade para participarem dos debates. A ameaça relacionada à inexperiência dos participantes em relação ao desenvolvimento de jogos digitais ou aos processos ágeis foi mitigada durante a seleção de participantes para o grupo focal: os participantes (todos oriundos do curso de bacharelado em Engenharia de Software pela UNIPAMPA) deveriam (1) ter cursado as disciplinas de Resolução de Problemas 5, ou (2) ter cursado a disciplina de Laboratório de Desenvolvimento de Jogos Digitais<sup>1</sup>, ou (3) ter experiência comprovada no desenvolvimento de jogos e em processos ágeis.

## 5.3 Lições do Capítulo

Neste capítulo apresentamos a técnica de validação do AgileGD através da execução do Grupo Focal, mostrando como foi o seu planejamento, execução, os resultados obtidos e as ameaçadas encontradas. Nesse capítulo foram identificadas algumas lições aprendidas, são elas:

- a técnica de validação chamada de grupo focal tem como objetivo validar o processo proposto neste trabalho, através da coleta de diversas opiniões de especialistas;
- a execução do grupo focal tem como objetivo forçar o debate dos especialistas com problemas e soluções encontradas na elaboração do processo, para o mesmo seja validado e refatorado.

<sup>1</sup> <http://cursos.unipampa.edu.br/cursos/engenhariadesoftware/files/2018/02/ppc-es-2018-1.pdf>

No capítulo a seguir apresentamos as considerações finais do trabalho, evidenciando a sua elaboração, as dificuldades encontradas, resultados e trabalhos futuros.



## 6 CONSIDERAÇÕES FINAIS

O principal objetivo deste trabalho foi propor um processo de desenvolvimento específico para jogos seguindo os princípios ágeis e respeitando uma série de requisitos identificados como essenciais em um processo desse domínio.

A Revisão Sistemática utilizada neste trabalho mostrou-se muito eficaz para a busca de trabalhos que relatam o uso de processos ágeis voltado ao desenvolvimento de jogos. Através da coleta e análise dos dados obtidos através da Revisão Sistemática, foi possível identificar os processos utilizados, os problemas e motivações relatados pelos autores para a criação, a adaptação ou utilização dos processos ágeis evidenciados. Mas a revisão sistemática demorou mais do que inicialmente planejado pelo fato da grande dificuldade de encontrar artigos relacionados ao tema do estudo. Com isso, tivemos que limitar a Revisão Sistemática a apenas uma base de dados. A análise dos trabalhos relacionados foi muito difícil, pois tivemos uma grande dificuldade de entender as diferenças entre os seus processos, devido aos autores não evidenciarem com clareza como estão utilizando, o que eles estão modificando e porque estão modificando os processos. Após superarmos essa dificuldade, os resultados da análise dos dados realizada pelos pesquisadores contribuiu para identificar dos requisitos que o AgileGD deveria conter para se diferenciar dos demais e suprir as necessidades relatadas.

Com isto, o processo foi pensado com a finalidade de cobrir os problemas relatados pelos autores que utilizam processos ágeis no desenvolvimento de jogos. O processo foi elaborado com base nos requisitos levantados através da análise dos artigos identificados na revisão sistemática, e modelado graficamente em BPMN.

O procedimento de validação do processo foi realizado através da técnica de Grupo Focal realizado com base na estrutura definida e apresentada neste trabalho. As principais contribuições obtidas com a execução do Grupo Focal, foram os resultados positivos e os negativos identificados através do questionário de participação, gravações e anotações dos relatos dos participantes na sua execução. Com esses resultados foi possível refatorar o processos com pontos que foram entendidos como importantes. Após a sua refatoração já podemos ter um indicativo inicial da validade do processo.

### 6.1 Trabalhos Futuros

Como trabalhos futuros para essa pesquisa, destacamos a possibilidade de realizar Grupos Focais adicionais com especialistas da área de processos ágeis de desenvolvimento de software e da área de desenvolvimento de jogos digitais separadamente para isolarmos as perspectivas sobre o processo. É possível também realizar um Survey<sup>1</sup> online com uma ampla amostra de desenvolvedores de jogos digitais para identificarmos a aceitação do uso desse processo por profissionais da indústria. Possivelmente também será realizado a

<sup>1</sup> A pesquisa Survey é um tipo de investigação quantitativa. Ela pode ser definida como uma forma de coletar dados e informações a partir de características e opiniões de grupos de indivíduos

análise das oportunidades de melhorias identificadas através do Grupo Focal já realizado, com o objetivo de aceitar pi rejeitar elas no processo.

## REFERÊNCIAS

- AL-AZAWI, R.; AYESH, A. Experience with aoab methodology in mobile computing module. In: IEEE. **Open Source Software Computing (OSSCOM), 2015 International Conference on**. Amman, 2015. p. 1–6. Citado 3 vezes nas páginas 34, 36 e 37.
- AL-AZAWI, R.; AYESH, A.; OBAIDY, M. A. Towards agent-based agile approach for game development methodology. In: IEEE. **Computer Applications and Information Systems (WCCAIS), 2014 World Congress on**. Hammamet, 2014. p. 1–6. Citado 3 vezes nas páginas 34, 36 e 37.
- ASUNCION, H. et al. Serious game development as an iterative user-centered agile software project. In: ACM. **Proceedings of the 1st International workshop on games and software engineering**. Waikiki, 2011. p. 44–47. Citado 3 vezes nas páginas 35, 36 e 37.
- BECK, K. Embracing change with extreme programming. **Computer**, IEEE, n. 10, p. 70–77, 1999. Citado 2 vezes nas páginas 26 e 27.
- BECK, K. et al. Manifesto for agile software development. 2001. Citado na página 25.
- BIFULCO, I. et al. The tie project: Agile development of a virtual world serious game on waste disposal. In: CITESEER. **DMS**. Fisciano, 2011. p. 204–209. Citado 3 vezes nas páginas 34, 36 e 37.
- BRIOL, P. **BPMN, the Business Process Modeling Notation Pocket Handbook**. [S.l.]: LuLu. com, 2008. Citado na página 24.
- DUNCAN, S. C. Minecraft, beyond construction and survival. **Well Played: a journal on video games, value and meaning**, ETC Press, v. 1, n. 1, p. 1–22, 2011. Citado na página 19.
- JENSEN, K.; VALENTE, A. Development of a memory training game. In: **Technologies of Inclusive Well-Being**. Berlin: Springer, 2014. p. 25–38. Citado 2 vezes nas páginas 33 e 36.
- KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. 2007. Citado na página 29.
- KONTIO, J.; BRAGGE, J.; LEHTOLA, L. The focus group method as an empirical tool in software engineering. In: **Guide to advanced empirical software engineering**. London: Springer, 2008. p. 93–116. Citado 2 vezes nas páginas 21 e 49.
- LIANA, I. et al. **INDIES : REALIDADE DOS DESENVOLVEDORES INDEPENDENTES**. 2014. Citado na página 20.
- MITRE-HERNÁNDEZ, H. A. et al. Decreasing rework in video games development from a software engineering perspective. In: **Trends and Applications in Software Engineering**. Cham: Springer, 2016. p. 295–304. Citado 3 vezes nas páginas 33, 36 e 37.
- NOVAK, J. **Desenvolvimento de Games (Tradução da 2a Edição Norte-Americana)**. São Paulo: Cengage Learning, 2010. Citado na página 23.

- OKTRIONO, K.; CHONG, H. Developing 3d action puzzle game application "the mechanic" using unity. In: IEEE. **e-Learning, e-Management and e-Services (IC3e), 2015 IEEE Conference on**. Melaka, 2015. p. 120–124. Citado 3 vezes nas páginas 33, 36 e 37.
- O'HAGAN, A. O.; COLEMAN, G.; O'CONNOR, R. V. Software development processes for games: a systematic literature review. In: SPRINGER. **European Conference on Software Process Improvement**. Berlin, 2014. p. 182–193. Citado 2 vezes nas páginas 20 e 29.
- PERES, A. L. et al. Methods and processes definitions for multiplatform social network games development with distributed teams. In: IEEE. **Games and Digital Entertainment (SBGAMES), 2011 Brazilian Symposium on**. Salvador, 2011. p. 189–195. Citado 3 vezes nas páginas 33, 36 e 37.
- PERSOON, M. **About the game**. 2011. Acessado em: 22 de setembro de 2018. Disponível em: <<http://www.minecraft.net/about.jsp>>. Citado na página 19.
- PETRILLO, F.; PIMENTA, M. Is agility out there?: agile practices in game development. In: ACM. **Proceedings of the 28th ACM International Conference on Design of Communication**. São Carlos, 2010. p. 9–15. Citado 3 vezes nas páginas 19, 33 e 36.
- PETRILLO, F. d. S. Práticas ágeis no processo de desenvolvimento de jogos eletrônicos. 2008. Citado na página 20.
- POLITOWSKI, C. et al. Are the old days gone?: A survey on actual software engineering processes in video game industry. In: ACM. **Proceedings of the 5th International Workshop on Games and Software Engineering**. New York, 2016. p. 22–28. Citado 2 vezes nas páginas 34 e 36.
- PRESSMAN, R. **Manifesto for agile software development**. New York: McGraw-Hill, 2009. Citado na página 27.
- PRESSMAN, R. **Engenharia de software: uma abordagem técnica**. Porto Alegre: AMGH Editora Ltda, 2011. Citado 2 vezes nas páginas 24 e 25.
- RABIN, S. Introdução ao desenvolvimento de games. **São Paulo: Cengage Learning**, 2012. Citado 2 vezes nas páginas 19 e 23.
- RODRIGUES, H. F.; MACHADO, L. d. S.; VALENÇA, A. M. G. Definição e aplicação de um modelo de processo para o desenvolvimento de serious games na área de saúde. In: **Proc. Congresso da Sociedade Brasileira de Computação-Workshop de Informática Médica**. João Pessoa: [s.n.], 2010. p. 1532–1541. Citado na página 19.
- SCHWABER, K.; SUTHERLAND, J. The scrum guide, 2013. 2016. Acessado em: 25 de setembro de 2018. Disponível em: <[http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf\(dostop28.4.2016\)](http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf(dostop28.4.2016))>. Citado 3 vezes nas páginas 25, 32 e 39.
- SMITH, T.; COOPER, K. M.; LONGSTREET, C. S. Software engineering senior design course: experiences with agile game development in a capstone project. In: ACM. **Proceedings of the 1st International Workshop on Games and Software Engineering**. Waikiki, 2011. p. 9–12. Citado 3 vezes nas páginas 35, 36 e 37.

SOMMERVILLE, I.; ARAKAKI, R.; MELNIKOFF, S. S. S. **Engenharia de software**. São Paulo: Pearson Prentice Hall, 2008. Citado 2 vezes nas páginas 24 e 26.

TAMBURRO, P. **GTA 5 is the Most Successful Entertainment Product in History**. 2018. Acessado em: 22 de setembro de 2018. Disponível em: <<https://www.gamerevolution.com/news/381213-gta-5-successful-entertainment-product-history>>. Citado na página 23.

TORRES-CARRIÓN, P. et al. Educational math game for stimulation of children with dyscalculia. In: SPRINGER. **International Conference on Information Theoretic Security**. Cham, 2018. p. 614–623. Citado 2 vezes nas páginas 32 e 36.

WAZLAWICK, R. **Engenharia de software: conceitos e práticas**. Rio de Janeiro: Elsevier Brasil, 2013. v. 1. Citado 2 vezes nas páginas 23 e 25.

WHITE, S. A. Introduction to bpmn: a tutorial. **IBM Corporation**, 2006. Citado na página 24.



## Apêndices



**APÊNDICE A – MATERIAL DE APOIO AO GRUPO FOCAL**

## **TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO**

**Título do projeto:** AgileGD: Um Processo Baseado em Abordagens Ágeis para o Desenvolvimento de Jogos

**Pesquisador responsável:** João Pablo da Silva da Silva

**Pesquisadores participantes:** Murilo da Rosa Bianchin

**Instituição:** Universidade Federal do Pampa – Unipampa

Você está sendo convidado para participar, como voluntário, de um grupo focal no trabalho de conclusão de curso (TCC) da Unipampa intitulado “AGILEGD: UM PROCESSO BASEADO EM ABORDAGENS ÁGEIS PARA O DESENVOLVIMENTO DE JOGOS”. Esse trabalho de conclusão de curso tem como objetivo elaborar um processo inspirado em abordagens ágeis para o desenvolvimento de jogos.

Você pode a qualquer momento pedir esclarecimentos sobre o TCC, o processo e os testes (informações coletadas, identificação dos participantes, outras pessoas envolvidas, etc.).

Você também poderá parar de participar a qualquer momento apenas avisando o pesquisador sem sofrer qualquer tipo de penalidade ou prejuízo.

Após ler e tirar suas dúvidas sobre as informações a seguir, se aceitar participar da pesquisa, assine no final deste documento, que tem duas cópias. Uma delas é sua e a outra será arquivada pelo pesquisador responsável.

### **O que você precisará fazer nos testes:**

1. Ir ao local do encontro com seus próprios meios.
2. Responder algumas perguntas depois de debater sobre o assunto com os outros participantes.
3. Conversar com os outros participantes sobre o tema estipulado a fim de mostrar seu ponto de vista e suas críticas a cada questão.

### **Riscos que você corre ao participar da pesquisa:**

1. Se irritar com outros participantes.
2. Se irritar pela demora em ser atendido ao pedir ajuda.

3. Ficar constrangido com alguma pergunta dos questionários.

**Benefícios da pesquisa:**

1. Ajudar o autor da monografia do TCC a elaborar um processo inspirado em abordagens ágeis com foco no desenvolvimento de jogos.

Participar dessa pesquisa não gera nenhum custo. Você também não receberá qualquer dinheiro ou ajuda financeira para participar da pesquisa ou chegar ao local do encontro.

Seu nome e outros dados serão mantidos em sigilo, e as informações coletadas na pesquisa (respostas dos questionários) serão guardadas pelos pesquisadores responsáveis. Os resultados poderão ser divulgados no texto final do TCC, em publicações ou outras formas de divulgação respeitando sempre o sigilo.

Nome do Participante da Pesquisa: \_\_\_\_\_

\_\_\_\_\_  
Assinatura do Participante da Pesquisa

Nome do Pesquisador Responsável: \_\_\_\_\_

\_\_\_\_\_  
Assinatura do Pesquisador Responsável

Local e data \_\_\_\_\_

## Questionário de Participação:

Nome: \_\_\_\_\_ Data: \_\_\_/\_\_\_/\_\_\_

### Problema 1:

**Q1:** Deve existir uma atividade responsável para elaborar o documento de design do jogo no processo.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_

**Q2:** Deve existir um documento de **design do jogo** no processo.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_

**Q3:** O nome artefato ser um nome genérico (**Documento de Design**), tendo em vista que esse documento poderá ser adaptável aos diferentes tipos de equipes e possuir diferentes formatos.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_

**Q4:** A solução apresentada resolve todos os problemas identificados.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

\_\_\_\_\_

## Questionário de Participação:

Nome: \_\_\_\_\_ Data: \_\_\_/\_\_\_/\_\_\_

### Problema 2:

**Q5:** Separar o time de desenvolvimento, manter um papel responsável pelo desenvolvimento do jogo (**codificação**) e um papel que englobe os outros papéis do desenvolvimento do jogo (**tarefas que não envolvem codificação**).

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

**Q6:** O nome do subprocesso é um nome genérico (**Realizar Atividades Especializadas**), tendo em vista que esse subprocesso poderá ser adaptável aos diferentes tipos de equipes e possuir diferentes tipos de atividades internas.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

**Q7:** A solução apresentada resolve todos os problemas identificados.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

## Questionário de Participação:

Nome: \_\_\_\_\_ Data: \_\_\_/\_\_\_/\_\_\_

### Problema 3:

**Q8:** Deve existir um teste de cada funcionalidade (**incremento**) que foi desenvolvida no sprint, com o público do teste de usabilidade sendo a própria equipe de desenvolvimento.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

**Q9:** Deve-se chamar a atividade de “**Realizar Teste com Jogador (Interno)**” e não chamar de versão “**Alfa**”, “**Pré Alfa**” e entre outros nomes por exemplo, deixando assim para a equipe decidir as características do teste de jogabilidade da forma que melhor se encaixa ao escopo do projeto.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

**Q10:** A solução apresentada resolve todos os problemas identificados.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

## Questionário de Participação:

Nome: \_\_\_\_\_ Data: \_\_\_/\_\_\_/\_\_\_

### Problema 4:

**Q11:** Deve existir um teste do **produto final (jogo)**, com o público do teste usabilidade sendo usuários fora do projeto de desenvolvimento.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

**Q12:** Deve se chamar de “Realizar Teste com Jogador (Externo)” e não chamar de versão “Beta”, “Open Beta”, “Release Candidate (RC)” e entre outras nomes por exemplo, deixando assim para a equipe decidir as características do teste de usabilidade da forma que melhor se encaixa ao escopo do projeto.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

**Q13:** A solução apresentada resolve todos os problemas identificados.

- Discordo fortemente
- Discordo
- Indiferente
- Concordo
- Concordo fortemente

Com base na resposta anterior, justifique sua resposta: \_\_\_\_\_

---

---

---

## Questionário de Nivelamento dos Participantes:

Nome: \_\_\_\_\_ Data: \_\_/\_\_/\_\_\_\_

**Q1:** Qual seu grau de escolaridade?\* \_\_\_\_\_

**Q2:** Qual a minha experiência no desenvolvimento de jogos?\*

- Muito Baixo ou Nenhum
- Baixa: Desenvolvi poucas vezes jogos
- Média: Desenvolvo ocasionalmente jogos
- Alta: Desenvolvo frequentemente jogos
- Muito Alta: Experiência industrial

**Q3:** Qual o meu conhecimento **teórico** em processos de desenvolvimento de software?\*

- Nenhum
- Baixo : Leitura de livro ou pesquisas na internet
- Moderado: Realização de algum curso ou palestra
- Alto: Experiência em cadeiras específicas
- Muito Alto: Experiência industrial

**Q4:** Qual o meu conhecimento **prático** em processos de desenvolvimento de software?\*

- Muito Baixo ou Nenhum
- Baixo: Utilizei poucas vezes processos de desenvolvimento de software
- Moderado: Utilizo ocasionalmente processos de desenvolvimento de software
- Alto: Utilizo frequentemente processos de desenvolvimento de software
- Muito Alto: Experiência industrial

**Q5:** Qual o meu conhecimento **teórico** em processos de desenvolvimento de jogos?\*

- Muito Baixo ou Nenhum
- Baixo: Leitura de livro ou pesquisas na internet
- Moderado: Realização de algum curso ou palestra
- Alto: Experiência em disciplinas específicas
- Muito Alto: Experiência industrial

**Q6:** Qual o meu conhecimento **prática** em processos de desenvolvimento de jogos?\*

- ) Muito Baixo ou Nenhum
- ) Baixo: Utilizei poucas vezes processos de desenvolvimento de jogos
- ) Moderado: Utilizo ocasionalmente processos de desenvolvimento de jogos
- ) Alto: Utilizo frequentemente processos de desenvolvimento de jogos
- ) Muito Alto: Experiência industrial

**Q7:** Qual o seu conhecimento **teórico** de processos ágeis?\*

- ) Muito Baixo ou Nenhum
- ) Baixo: Leitura de livro ou pesquisas na internet
- ) Moderado: Realização de algum curso ou palestra
- ) Alto: Experiencia em disciplinas específicas
- ) Muito Alto: Experiência industrial

**Q8:** Qual o seu conhecimento **prático** de processos ágeis?\*

- ) Muito Baixo ou Nenhum
- ) Baixo: Utilizei poucas vezes processos ágeis
- ) Moderado: Utilizo ocasionalmente processos ágeis
- ) Alto: Utilizo frequentemente processos ágeis
- ) Muito Alto: Experiência industrial

**Q9:** Qual o seu conhecimento **teórico** com Scrum?\*

- ) Muito Baixo ou Nenhum
- ) Baixo: Leitura de livro ou pesquisas na internet
- ) Moderado: Realização de algum curso ou palestra
- ) Alto: Experiencia em disciplinas específicas
- ) Muito Alto: Experiência industrial

**Q10:** Qual o seu conhecimento **prático** com Scrum?\*

- ) Muito Baixo ou Nenhum
- ) Baixo: Utilizei poucas vezes o Scrum
- ) Moderado: Utilizo ocasionalmente o Scrum
- ) Alto: Utilizo frequentemente o Scrum
- ) Muito Alto: Experiência industrial

**Q11:** Qual o seu conhecimento **teórico** em Extreme Programming?\*

- ) Muito Baixo ou Nenhum
- ) Baixo: Leitura de livro ou pesquisas na internet
- ) Moderado: Realização de algum curso ou palestra
- ) Alto: Experiencia em disciplinas específicas
- ) Muito Alto: Experiência industrial

**Q12:** Qual o seu conhecimento **prático** em Extreme Programming?\*

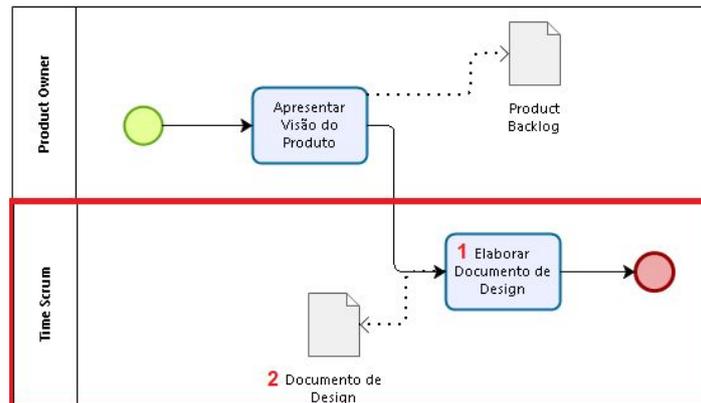
- ) Muito Baixo ou Nenhum
- ) Baixo: Utilizei poucas vezes processos de desenvolvimento de jogos
- ) Moderado: Utilizo ocasionalmente processos de desenvolvimento de jogos
- ) Alto: Utilizo frequentemente processos de desenvolvimento de jogos
- ) Muito Alto: Experiência industrial

## Material de Suporte para o Problema 1:

### Descrição do problema

- Muitos autores relataram que deve existir um documento de **design do jogo** nos processos de desenvolvimento de jogos;
- A base do processo é o Scrum, pois se mostrou a melhor opção de base do processo, e sabemos que o Scrum não é específico para jogos, logo não possui uma atividade específica para o **design do jogo**;
- **Problema: Como incluir o documento de design nesse novo processo?**

### Solução



A solução foi incluir no processo a atividade Elaborar Documento de Design na qual gera um artefato de saída chamado Documento de Design.

### Manual de utilização das atividades e artefatos envolvidos no problema

1. A atividade **Elaborar Documento de Design** é a atividade responsável por reunir os integrantes do projeto (Time Scrum) em uma reunião no qual possui o propósito de definir e levantar informações importantes para a realização do desenvolvimento do jogo, e que tenha como saída dessa reunião um documento de design do jogo.

2. O artefato gerado na atividade Elaborar Documento de Design é o **Documento de Design**, no qual é o documento responsável por conter informações de design do jogo, contudo, o mesmo não possui um template específico. O objetivo do Documento de Design de não possuir um template específico é que o mesmo possa ser adaptável aos diferentes tipos de projetos e equipes, tornando-se assim flexível para cada equipe de projeto definir a sua estrutura de documento que se adeque melhor aos seus projetos, porém o documento deve possuir informações referentes a algumas áreas consideradas necessárias para a maioria dos projetos, lembrando que não precisa ter seções com os respectivos nomes e sim conter informações sobre as respectivas áreas, são elas:

- **Conceito:** Possui informações como: nome do jogo, estilo do jogo, público alvo, história, etc.
- **Especificações técnicas:** Possui informações como: hardware, requerimentos de software, hardware mínimo, sistema operacional, etc.
- **Especificações do jogo:** Possui informações como: número de fases, níveis de dificuldade, número de jogadores, personagens, itens de jogo, itens de cenários, etc.
- **Dispositivos de entrada:** Possui informações como: suporte para mouse, dispositivos de entrada, definição de teclas e botões, etc.
- **Design gráfico e arte:** Possui informações como: definição de fases, definição visual, etc.
- **Sonorização:** Possui informações como: música nos menus, músicas nas fases, efeitos, etc.
- **Desenvolvimento:** Possui informações como: tempo de desenvolvimento, alocação de pessoal, cronograma, metas, etc.

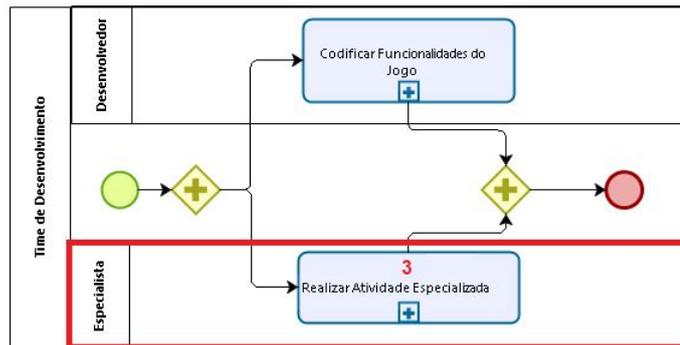
Exemplo de **Documento de Design** que poderá ser utilizado em um projeto está disponibilizado em **anexo**.

## Material de Suporte para o Problema 2:

### Descrição do problema

- Muitos autores relatam, que as equipes de desenvolvimento de jogos acabam sendo bem **diversificadas** e os processos acabam não dando o suporte necessário para a equipe nessa situação;
- Os autores relatam também que muitos processos acabam separando as equipes de desenvolvimento, por não terem espaço para designer, artistas e entre outros papéis;
- **Problema: Como resolver esses problemas relatados no novo processo?**

### Solução:



A solução foi criar um papel chamado de Especialista que engloba todos os outros papéis que não são ligados a codificação do jogo e incluir uma atividade com sub atividades adaptáveis chamado Realizar Atividade Especializada.

### Manual de utilização/adaptação das atividades envolvidas no problema

**3. A atividade Realizar Atividades Especializadas** apresenta as atividades que serão realizadas diariamente pelos especialistas. As atividades internas dessa atividade serão definidas conforme a necessidade de cada equipe, ou seja, o objetivo é que as atividades internas possam ser adaptáveis aos diferentes projetos de desenvolvimento de jogos.

- **A solução para adaptar essa atividade** é que a equipe escolha as atividades internas da atividade Realizar Atividades Especializadas da forma que dê suporte aos diferentes tipos de equipes do projeto, a equipe deve seguir algumas regras para adaptar, são elas: só podem ser atribuídas atividades internas que **não sejam envolvidas com a codificação** do jogo ou que o papel envolvido seja o **desenvolvedor**, com isso, o objetivo é que todas as atividades ligadas ao desenvolvimento do jogo e que não sejam atividades de codificação, sejam atribuídas a esse subprocesso, o mesmo se enquadra aos papéis.
- **Exemplo de uso:** Suponhamos que o projeto contenha três especialistas: um é responsável pelo desenvolvimento do áudio do jogo (**designer de som**), um pelo design de personagem (**artista**) e o outro para modelagem do ambiente (**artista**). A **figura 1** apresenta como ficaria essas atividades internas do subprocesso Realizar Atividade Especializada adaptada para as necessidades da equipe do projeto específica desse exemplo.

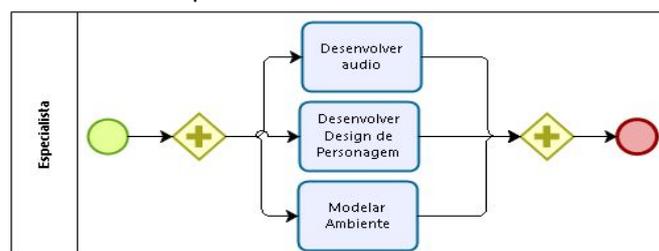


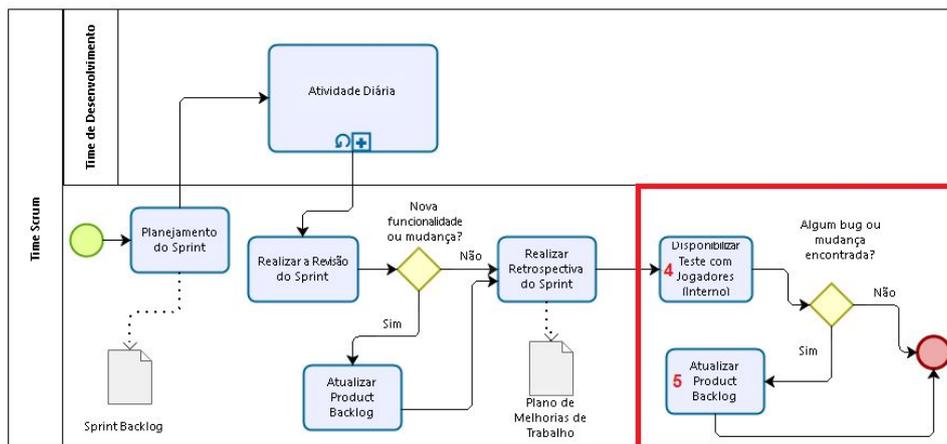
Figura 1: Exemplo do subprocesso Realizar Atividade Especializada customizada.

## Material de Suporte para o problema 3:

### Descrição do problema

- No desenvolvimento de um jogo deve-se realizar testes de jogabilidade para cada funcionalidade ou grupo de funcionalidades implementadas (**incremento**), esses testes são realizados com a própria equipe do projeto, tendo em vista que o projeto ainda possui muitos bugs e necessidades de melhorias;
- Alguns projetos chamam esse teste de versão **pré Alfa, Alfa** e entre outras variações;
- **Problema: Como incrementar o teste de jogabilidade ainda na fase de desenvolvimento do projeto?**

### Solução



A solução foi criar uma atividade chamada Disponibilizar Teste com Jogadores (Interno) no final do sprint, com o objetivo de obter feedback sobre bug e/ou melhorias no projeto ainda no sprint, se for identificado algum bug e/ou melhoria o product backlog é atualizado para que no próximo sprint sejam corrigidos.

### Manual de utilização das atividades envolvidas no problema

**4.** A atividade **Disponibilizar Teste com Jogadores (Interno)** é responsável por planejar e disponibilizar o teste da(s) jogabilidade(s) do incremento desenvolvido no sprint, O teste do incremento será realizado com os integrantes do próprio projeto, tendo em vista que ainda possuem muitos bugs e melhorias nas funcionalidades do jogo. O objetivo de pôr um nome genérico nessa atividade e não por “Pré Alfa” ou “Alfa” que geralmente são utilizados por exemplo, é a de que essa atividade pode ser adaptável aos diferentes projetos de desenvolvimento de jogos, assim, não seguirá uma estrutura de lançamento e execução de testes previamente definidas, ficando a critério da equipe do projeto definir a melhor forma de realização.

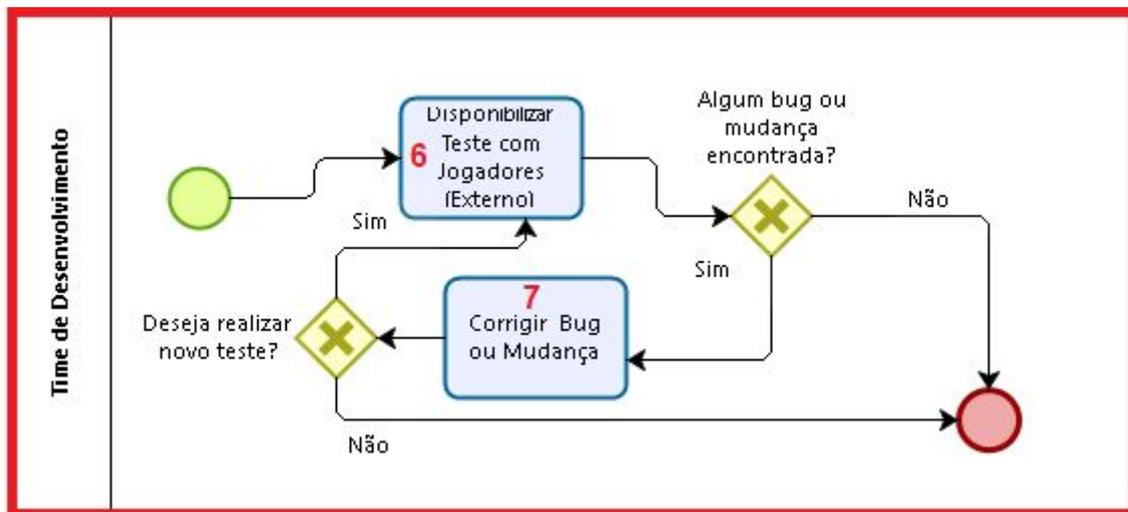
**5.** A atividade **Atualizar Product Backlog** é executada sempre após a realização da atividade Disponibilizar Teste com Jogadores (Interno), se for identificado algum bug e/ou mudança no incremento ou no projeto, ela tem como atualizar o product backlog do projeto com as refatorações necessárias, para que no próximo sprint elas sejam refatoradas.

## Material de Suporte para o problema 4

### Descrição do problema

- No desenvolvimento de um jogo deve-se realizar testes do jogo **todo desenvolvido**, esses testes são realizados com o público externo e possui como objetivo encontrar bugs, melhorias e/ou dar um feedback a respeito do produto (**jogo**);
- Alguns projetos chamam esse teste de versão **Closed Beta, Beta, Open Beta, Release Candidate (RC), entre outros**;
- **Como incrementar o teste de jogabilidade na fase do pós projeto do jogo?**

### Solução



A solução foi criar um subprocesso Pré Lançamento do jogo na fase de Pós Projeto e nela conter uma atividade chamada Disponibilizar Teste com Jogadores (Externo)

### Manual de utilização das atividades envolvidas no problema

**6.** A atividade **Disponibilizar Teste com Jogadores (Externo)** é responsável por planejar e disponibilizar o teste de usabilidade do produto desenvolvido (jogo), o teste de usabilidade do jogo será realizado com o público externo, tendo em vista que alguns bugs ou melhorias a equipe já possui conhecimento, esse teste pode ser realizado quantas vezes e da forma que a equipe de projeto definir. O objetivo de pôr um nome genérico nessa atividade e não por **Closed Beta, Beta, Open Beta, Release Candidate (RC)** e **entre outros**, que nos quais geralmente são utilizados na indústria, é a de que essa atividade pode ser adaptável aos diferentes projetos de desenvolvimento de jogos, não necessariamente seguindo uma estrutura de lançamento e execução de testes previamente definida, ficando a critério da equipe do projeto definir a melhor forma de realização.

**7.** A atividade **Corrigir Bug ou Mudança** é executada sempre depois de realizar a atividade Disponibilizar Teste com Jogadores (Externo), e se for identificado algum bug e/ou mudança no jogo, ela tem como objetivo de refatorar os problemas e/ou melhorias encontradas, a fim de atualizar a versão de teste com versões de atualização ou finalizar a versão de teste, se for desejado realizar novo teste é refeito o ciclo de teste até quando a equipe desejar.



## APÊNDICE B – GUIA DE ADAPTAÇÃO

## Guia de adaptação da atividade Realizar Atividade Especializada

Esse guia tem como objetivo auxiliar os usuário na adaptação do subprocesso da atividade Realizar Atividades Especializadas, o guia possui uma descrição com uma explicação detalhada e de fácil entendimento com exemplos e um processo.

**A atividade Realizar Atividades Especializadas** apresenta as atividades que serão realizadas diariamente pelos especialistas. As atividades internas dessa atividade serão definidas conforme a necessidade de cada equipe, ou seja, o objetivo é que as atividades internas possam ser adaptáveis aos diferentes projetos de desenvolvimento de jogos.

- **A solução para adaptar essa atividade** é que a equipe escolha as atividades internas da atividade Realizar Atividades Especializadas da forma que dê suporte aos diferentes tipos de equipes do projeto, a equipe deve seguir algumas regras para adaptar, são elas: só podem ser atribuídas atividades internas que **não sejam envolvidas com a codificação** do jogo ou que o papel envolvido seja o **desenvolvedor**, com isso, o objetivo é que todas as atividades ligadas ao desenvolvimento do jogo e que não sejam atividades de codificação, sejam atribuídas a esse subprocesso, o mesmo se enquadra aos papéis.
- **Exemplo de uso:**
  - **Cenário 1:** Suponhamos que o projeto contenha três especialistas: um é responsável pelo desenvolvimento do áudio do jogo (**designer de som**), um pelo design de personagem (**artista**) e o outro para modelagem do ambiente (**artista**). A **figura 1** apresenta como ficaria essas atividades internas do subprocesso Realizar Atividade Especializada adaptada para as necessidades da equipe do projeto específica desse exemplo.

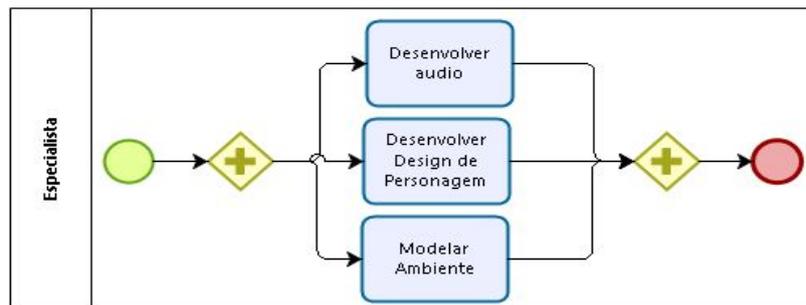


Figura 1: Exemplo do subprocesso Realizar Atividade Especializada customizada.

- **Cenário 2:** Suponhamos que o projeto contenha dois especialistas: um é responsável pelo desenvolvimento da arte dos personagens (**designer**) e outro pela modelagem do ambiente (**artista**). A figura 2 apresenta como ficaria essas atividades internas do subprocesso Realizar Atividade Especializada adaptada para as necessidade da equipe do projeto específica desse exemplo.

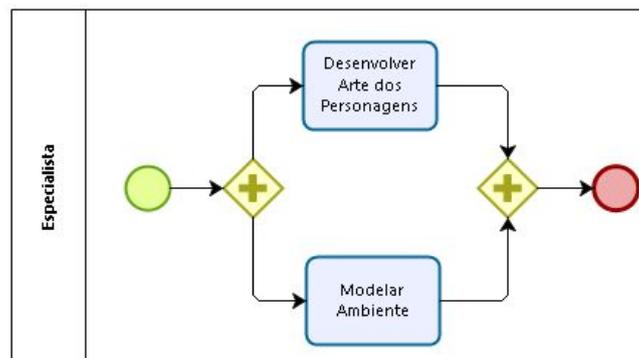


Figura 2: Exemplo do subprocesso Realizar Atividade Especializada customizada.

## Processo para adaptar a atividade Realizar Atividade Especializada

Para auxiliar ainda mais o usuário a adaptar essa atividade, elaboramos um processo para instanciar essa adaptação.

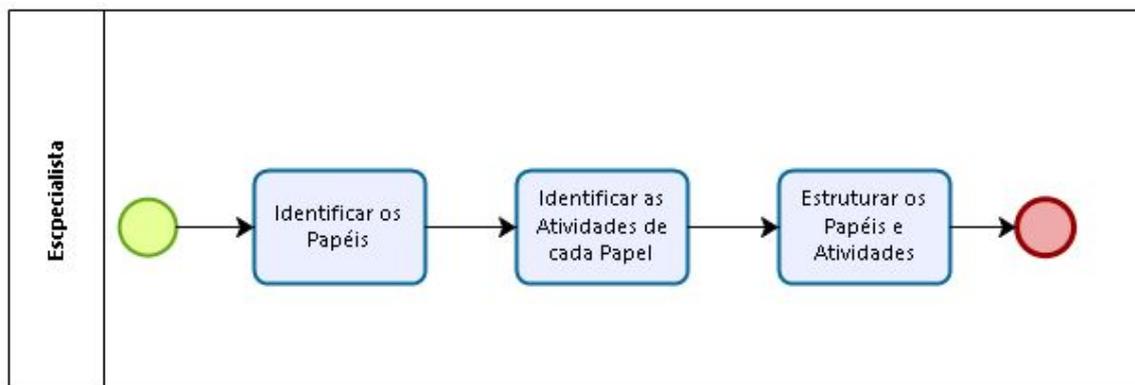


Figura 3: Processo para instanciar a adaptação da atividade Realizar Atividade Especializada

Na atividade **Identificar os Papéis** será realizada uma reunião com todos integrantes do projeto, com o objetivo identificar os papéis que se encaixam nos requisitos de especialistas, ou seja, que são papéis que possuam atividades que não sejam envolvidas com a codificação do jogo ou que o papel envolvido não seja o papel de desenvolvedor, após essa identificação a equipe executará a próxima atividade.

A atividade **Identificar as Atividades de cada Papel**, que é realizada após a identificação dos papéis que se encaixam no escopo do papel Especialista, será identificado as atividades que esses papéis irão executar no projeto de desenvolvimento do jogo, com o objetivo que a equipe estruture suas atividades diárias dos respectivos papéis dentro do time de desenvolvimento.

Por fim será realizada a atividade **Estruturar Papéis e Atividades** no subprocesso é responsável por unir os papéis identificados e as atividades de cada papel em uma de realização das atividades para que possam ser atribuídas ao subprocesso Realizar Atividades Especializadas e conseqüentemente ser executada em paralelo diariamente com a atividade Realizar Desenvolvimento do Jogo, a equipe definirá a melhor forma de organizar a estrutura do processo para conter os papéis e atividades identificadas.