

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

**UMA ABORDAGEM GAMIFICADA PARA A
GERÊNCIA DE RISCOS**

DISSERTAÇÃO DE MESTRADO

GABRIEL BELINAZO

Santa Maria, RS, Brasil

2019

Gabriel Belinazo

**UMA ABORDAGEM GAMIFICADA PARA O
GERENCIAMENTO DE RISCOS**

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Ciência da Computação (PGCC), Área de Concentração em Computação Aplicada, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do título de **Mestre em Ciência da Computação**

Orientador: Prof^ª. Dr^ª. Lisandra Manzoni Fontoura

Santa Maria, RS, Brasil

2019

Belinazo, Gabriel

Uma abordagem gamificada para a gerência de riscos
/Gabriel Belinazo - 2019.

73 p.; 30 cm

Orientadora: LISANDRA MANZONI FONTOURA

Dissertação (mestrado) - Universidade Federal de Santa
Maria, Centro de Tecnologia, Programa de Pós-Graduação em
Ciência da Computação, RS, 2019

1. GERENCIA DE RISCOS 2. PROJETO DE SOFTWARE
3. GAMIFICAÇÃO I. MANZONI FONTOURA, LISANDRA II. Título.

Gabriel Belinazo

**UMA ABORDAGEM GAMIFICADA PARA O
GERENCIAMENTO DE RISCOS**

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Ciência da Computação (PGCC), Área de Concentração em Computação Aplicada, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do título de **Mestre em Ciência da Computação**

Aprovado em 29 de agosto de 2019.

Lisandra Manzoni Fontoura, Dr.
(Presidente/Orientador)

Cristiano Tolfo, Dr. (UNIPAMPA)

Giliane Bernardi, Dr. (UFSM)

Santa Maria, RS.
2019

AGRADECIMENTOS

Sempre, ao ler uma monografia, dissertação ou tese, início a leitura pela seção de agradecimentos, pois para mim dá um vislumbre de como o pesquisador se relaciona com o mundo, e, por conseguinte, como o entende. Dessa forma, para mim é também fundamental agradecer.

Em primeiro lugar, agradeço a Universidade Federal de Santa Maria, braço do estado, executor de políticas de ensino, que tem sido a instituição de referência durante toda a minha vida. Por morar em seus arredores, não sei dizer quando foi a primeira vez que visitei o seu campus em Camobi. No entanto, sei dizer que em 2004 ingressei no então Colégio Agrícola de Santa Maria - CASM, vinculado a UFSM, e ali permaneci até 2006. No ano de 2007 ingressei no curso de Letras Inglês e Literaturas de Língua Inglesa, curso que interrompi em 2009, mesmo ano em que ingressei no curso de Sistemas de Informação, na sua primeira turma, ofertado a partir do programa REUNI, do qual me formei em 20 de setembro de 2013. E por fim, retornei para cursar a pós-graduação stricto sensu, sem a UFSM minha história, assim como a de tantos outros seria muito diferente.

À minha orientadora, Prof^a. Dr^a Lisandra, que me acompanhou durante esse processo, agradeço sua disposição, compreensão, confiança e amizade.

Aos colegas de programa, que durante o curso do mestrado pudemos compartilhar boas conversas.

Em especial a Darciele de Souza por dividir as angústias, os desesperos, os choros e os risos desde 2009. E a Mariana Kegler Lorentz por evitar assuntos pesados, e constante companhia para reclamações de sono e fome.

Agradeço aos meus pais pela compreensão das ausências, dos irritumes, dos desesperos, e também das felicidades. E in memoriam agradeço a minha vó Elvira, gostaria de nunca estar cansado assim como ela nunca estava, nem nos momentos finais.

Agradeço ao IFFar por me conceder a dispensa de horas e o incentivo para que buscasse a qualificação profissional, em especial nas pessoas do prof. Carlos Eugênio Rodrigues Balsemão e do prof. Thiago Santi Bressan, que enquanto meus chefes imediatos incentivaram-me. Além, é claro, dos amigos Taigra, Fábio, Melissa, Derleida e Luciele, que em todos os dias demonstravam sua paciência e apoio.

Aos amigos Érico, Elber, Gabriel, Camila, Iochane, obrigado por compartilharem risadas, conversas, conselhos.

Aos demais amigos, familiares, professores que sempre colaboraram para o meu desenvolvimento enquanto ser humano, profissional e pesquisador.

RESUMO

UMA ABORDAGEM GAMIFICADA PARA O GERENCIAMENTO DE RISCOS

AUTOR: GABRIEL BELINAZO

ORIENTADORA: LISANDRA MANZONI FONTOURA

Em todo e qualquer projeto é fundamental considerarmos a possibilidade de existirem situações adversas ao projeto. Essas situações, que podem ou não ocorrerem, na prática, são chamadas de riscos. Em projetos de software não é diferente, e, com isso, o prazo, o custo e a necessidade do cliente podem não ser atendidos se não tivermos claro quais situações podem impactar negativamente no projeto. A complexidade inerente do processo de gerenciamento de riscos, somado as suas características de planejamento o tornam desinteressante diversas vezes. Além da motivação, para que tenhamos um melhor monitoramento e controle da gestão de riscos, podemos inserir elementos de jogos. Por conta disso, essa dissertação apresenta uma abordagem para gamificação do processo de gerenciamento de riscos. Essa abordagem gamificada consiste em fases nas quais o gerente configura o jogo, organizando riscos importantes e sua probabilidade e impacto no processo e selecionando ou criando ações que podem ser tomadas para mitigar ou evitar esses riscos, compondo o plano de gerenciamento de risco. A abordagem objetiva criar uma rotina de gerenciamento qualitativo de riscos, a fim de popularizar tal atividade, dada sua importância estratégica para as organizações. Ao fim da execução do projeto, é gerado também um relatório em que constam as atividades desenvolvidas e o entendimento de riscos no projeto, para que seja possível ampliar o conhecimento de riscos da equipe, bem como melhorar o processo de desenvolvimento de software.

Palavras-chave: gamificação; gerenciamento de riscos; gerenciamento de projeto de software.

ABSTRACT

A GAMIFIED APPROACH TO THE RISK MANAGEMENT

AUTHOR: GABRIEL BELIANZO

ADVISER: LISANDRA MANZONI FONTOURA

In any project, it is essential that the possibility of adverse situations to the project is considered as a factor. These situations that may or may not occur in practice are called risks. In software projects it is no different. Deadline, cost and customer needs may not be met if which situations may negatively impact the project are not clear for the stakeholders. The inherent complexity of the risk management process plus its planning characteristics make it uninteresting several times. In addition to motivation, in order to have better monitoring and control of risk management we can insert game elements. Because of this, this dissertation presents an approach to gamification of the risk management process. This gamified approach consists of phases in which the manager configures the game, organizing key risks and their probability and impact on the process, and selecting or creating actions that can be taken to mitigate or avoid those risks, composing the risk management plan. The approach aims to create a qualitative risk management routine in order to popularize such activity, given its strategic importance for organizations. At the end of the project execution, a report is also generated containing the activities developed and the understanding of risks in the project, so that it is possible to expand the knowledge of risks of the team, as well as improve the software development process.

Key words: gamification; risk management; software project management.

LISTA DE FIGURAS

Figura 1 - Modelo Espiral.....	24
Figura 2 – Processo de criação do personagem.....	34
Figura 3 - Ficha de personagem	35
Figura 4 – Metamodelo da abordagem gamificada de gerenciamento de projetos	37
Figura 5 - Metamodelo da abordagem de gerenciamento de riscos gamificado	39
Figura 6 - Ficha de monstro em D&D.....	41
Figura 7 - Ficha de monstro.....	42
Figura 8 - Riscos.....	43
Figura 9 - Tela para adição de tarefas.....	43
Figura 10 - Tela de cadastro de missões.....	44
Figura 11 - Novo usuário jogador do sistema.....	49
Figura 12 - Ficha de Personagem	50
Figura 13 – Maiores desafios de projeto	54
Figura 14 – Áreas de falha de projetos	54

LISTA DE TABELAS

Tabela 1 - Relação entre D&D e a abordagem gamificada	45
---	----

LISTA DE ABREVIATURAS E SIGLAS

CMMI	<i>Capability Maturity Model Integration</i>
DM	<i>Dungeon Master</i>
D&D	<i>Dungeons & Dragons</i>
ISO	<i>International Organization for Standardization</i>
MPS.Br	Melhoria de Processo do Software Brasileiro
PEEnSo	Pesquisa em Engenharia de Software
PMBOK	<i>Project Management Body of Knowledge</i>
PMI	<i>Project Management Institute</i>
PO	<i>Product Owner</i>
RPG	<i>Role Playing Game</i>
RUP	<i>Rational Unified Process</i>
SEI	<i>Software Engineering Institute</i>
XP	<i>eXtreme Programming</i>

LISTA DE ANEXOS

Anexo A – Histórias de usuário do sistema	X
Anexo B – Listagem de riscos pré-carregados no sistema	XI
Anexo C – Questionário	XII

SUMÁRIO

1. INTRODUÇÃO	19
1.1 DEFINIÇÃO DO PROBLEMA	21
1.2 ESCOPO E CONTRIBUIÇÕES DA PESQUISA	21
1.3 ORGANIZAÇÃO DA DISSERTAÇÃO.....	22
2. REVISÃO DE LITERATURA	23
2.1 GERENCIAMENTO DE RISCOS EM PROJETOS DE SOFTWARE.....	23
2.3 GAMIFICAÇÃO	29
3. TRABALHOS RELACIONADOS	32
4. UMA ABORDAGEM GAMIFICADA PARA O GERENCIAMENTO DE PROJETOS	34
5. A ABORDAGEM GAMIFICADA PARA O GERENCIAMENTO DE RISCOS	37
6. VALIDAÇÃO	48
6.1 VALIDAÇÃO DE MODELOS TEÓRICOS	48
6.2 QUESTIONÁRIOS A ESPECIALISTAS.....	53
6.3 ESTUDO DE CASO.....	57
7. CONSIDERAÇÕES FINAIS	59
REFERÊNCIAS	61
ANEXO A – Histórias de usuário do sistema	63
ANEXO B – Riscos pré-carregados no sistema.....	67
ANEXO C – QUESTIONÁRIO	72

1. INTRODUÇÃO

Em qualquer processo de software, podem ocorrer problemas concernentes a quaisquer etapas do processo ou envolvidos, sejam estimativas, requisitos, ferramentas, pessoas, organizações ou tecnologias. Segundo o *Project Management Body of Knowledge - PMBoK* (PMI, 2017), um projeto é bem-sucedido quando satisfaz as necessidades do cliente, é entregue dentro do prazo e com o custo e a qualidade estimados. Riscos fazem referência a possibilidades futuras de eventos que possam atingir o projeto de forma global, dessa forma riscos envolvem escolhas e incertezas. Para tanto, é necessário que se gerenciem os problemas que podem surgir no curso do projeto. “Os riscos podem ser aceitos, gerenciados, mitigados, evitados, mas não podem ser negados” (LATHAM, 1994).

Os processos tradicionais de gerenciamento de risco propõem que sejam identificados os riscos de um projeto, avaliados sua probabilidade de ocorrência e seu impacto, gerando uma lista e dados quantitativos, que são organizados em forma de matriz, a fim de classificar os riscos identificados por ordem de impacto e probabilidade. Para que, assim, a equipe possa estabelecer um plano de gerenciamento que envolva atividades e práticas a fim de evitar ou mitigar os riscos de maior impacto e probabilidade (PRESSMAN, 2016).

A gestão de riscos em projetos de software sempre foi uma preocupação, embora apenas tenha ganhado a devida atenção com a proposta de ciclo de vida de Boehm (BOEHM, 1991) nos anos 90. No modelo de Boehm (1991), é proposto que o gerenciamento de riscos seja conduzido, dentro do processo de software, da seguinte forma, a cada iteração, após serem definidos os objetivos da iteração, bem como seus requisitos. Os riscos, nessa abordagem, devem ser identificados, analisados e priorizados, fases que compõem a avaliação de risco; após a avaliação de risco, seguem-se as fases que compõem o controle de risco que são: planejamento de risco, a resolução de riscos e o monitoramento.

Nesse modelo, a identificação de riscos produzirá uma lista de riscos relativos ao contexto do projeto. De posse da lista, passa-se a análise dos riscos, em que são considerados as perdas relevantes e os resultados não satisfatórios que podem advir desses riscos, nessa fase, podem-se tomar dados históricos de projetos anteriores. Após os riscos identificados e analisados, procede-se a priorização, considerando-se, principalmente, o impacto e a probabilidade. Por fim, passa-se ao conjunto de fases de controle de risco, iniciando com o planejamento do controle de riscos, que se trata do momento em que os riscos priorizados ganham um plano de ação, sem que isso envolva muito tempo ou esforço, de modo a preparar

de forma efetiva para o controle do risco. Na fase de resolução de risco se dá a implantação de todos os elementos planejados. E, por fim, a fase de monitoramento de riscos, que dá ao projeto como um todo um modo para garantir que o gerenciamento de riscos é controlado (BOEHM, 1991). Desde o modelo em espiral, o gerenciamento de riscos aumentou em complexidade.

Modelos de processos, como os propostos pelo PMI, pela ISO 31000 e pelo CMMI têm sido implementados em mais larga escala, dada as necessidades mercadológicas de certificação. Fato esse que cria uma complexidade ainda maior, que somada à complexidade inerente do gerenciamento de riscos tende a diminuir a motivação da equipe, causando falhas de gerenciamento de riscos durante a execução do processo de gerenciamento em projetos de software (WALLACE *et al.*, 2004). Uma pesquisa conduzida por Wallace *et al.* (2004) apontou que considerando os mais diversos níveis de riscos em projetos, mesmo em projetos com baixo risco, há alta complexidade inerente ao gerenciamento de riscos.

Segundo WAGNER e RUHE (2018), embora os fatores de produtividade de uma equipe sejam recorrentemente relacionados à complexidade e tamanho do produto, há outros aspectos a se considerar, uma vez que cerca de 30% do tempo de um desenvolvedor está envolvido em atividades que não tem cunho técnico.

Medeiros (2015) afirma que o uso de elementos de jogo "não só pode ajudar a envolver as equipes em seu trabalho, mas também pode ajudar a monitorar, controlar e melhorar seu desenvolvimento". A gamificação é o uso de elementos de design de jogos em contextos diferentes, a fim de mudar o comportamento, aumentar o envolvimento e a motivação das pessoas na realização de atividades.

Dubois e Tamburrelli (2013) afirmam que a gamificação foi inicialmente adotada para aumentar o engajamento de consumidores que recebem pontos e medalhas no uso de serviços. Nos últimos anos, a gamificação tem sido sistematicamente adotada como uma possível solução para o desinteresse dos alunos no desenvolvimento de cursos e atividades educativas e, por isso, tem surgido como uma tendência em vários estudos. Existem muitos motivos pelos quais a gamificação rapidamente se tornou uma palavra da moda, entre eles, Kumar (2013) destaca: a natureza mutável da informação, entrada na força de trabalho dos nativos da era digital - uma nova geração que cresceu jogando videogames, a ampla adoção das mídias sociais, a adoção de tecnologias móveis.

Portanto, esse estudo tem como objetivo aplicar um modelo efetivo de gerenciamento de riscos, utilizando uma abordagem *gamificada*, a fim de aumentar a motivação das equipes de desenvolvimento. Isso aplicado a um contexto de desenvolvimento da abordagem de

gerenciamento de projetos gamificada desenvolvida por Souza (2019, no prelo), no Laboratório de Computação Aplicada da Universidade Federal de Santa Maria.

1.1 DEFINIÇÃO DO PROBLEMA

É sabido que o gerenciamento de riscos é uma tarefa complexa. Uma pesquisa, que considerou os mais diversos níveis de riscos em projetos, revelou que mesmo os projetos de baixo risco têm um alto nível de complexidade para a correta gestão de riscos (MENEZES JÚNIOR et al, 2018). Dessa forma, nota-se que a complexidade da correta gestão de riscos implica na falha de projetos pela falta de compreensão dos riscos envolvidos (SARIGIANNIDIS e CHATZOGLU, 2013).

Especialistas neste campo alegam que o risco relacionado a projetos de desenvolvimento de software primeiro deve ser definido, para que, em seguida, possa ser tratado durante o processo de desenvolvimento (WALLACE *et al.*, 2004). Dessa forma, por se tratar de uma tarefa bastante complexa, uma vez que ela engloba diversos aspectos de predição, planejamento e outras atividades burocráticas, há uma queda na motivação dos desenvolvedores. Para Wagner e Ruhe (2018), estudos sobre a produtividade e motivação enfocam atividades puramente técnicas em sua maioria. Nesse aspecto, essa pesquisa objetiva pesquisar formas de aumentar a motivação nas atividades relacionadas ao gerenciamento de riscos de projeto, uma vez que essas atividades não são puramente técnicas, envolvendo aspectos gerenciais e preditivos.

1.2 ESCOPO E CONTRIBUIÇÕES DA PESQUISA

Nessa dissertação definiu-se uma proposta de abordagem gamificada ao gerenciamento de riscos em projetos de software ao longo da execução de um projeto de software gamificado. Em vista disso, as contribuições desse trabalho incluem:

- Elaboração de um metamodelo para estruturar os elementos de gamificação e atividades relacionadas a gerenciamento de riscos em processos de software;

- Identificação de relacionamento entre práticas do processo gamificado com modelos de processos de referência;
- Seleção de um *checklist* para a gestão de riscos em projetos de software;
- Seleção de um conjunto de boas práticas para o gerenciamento de riscos ativo;

1.3 ORGANIZAÇÃO DA DISSERTAÇÃO

Esse trabalho encontra-se organizado nos seguintes capítulos: 2. Revisão de Literatura, que se subdivide em 2.1 Gerenciamento de Riscos em Projetos de Software e 2.3 Gamificação; A partir desses conhecimentos, o capítulo 3. Trabalhos Relacionados analisa trabalhos sobre o tema. A seguir, escrutina-se a abordagem Ranger no capítulo 4. Ranger: Uma abordagem Gamificada para Gerenciamento de Riscos. Para que no capítulo 5. A abordagem gamificada para gerenciamento de riscos é proposta. Segue-se para a apresentação da validação no capítulo 6 e em suas subdivisões. Por fim são apresentadas as considerações finais no capítulo 7, e as referências no capítulo 8.

2. REVISÃO DE LITERATURA

Esse capítulo trata da revisão de literatura necessária a compreensão do tema abordado na proposta de uma abordagem gamificada. Dessa forma será apresentado em 2.1 Gerenciamento de Riscos, 2.2 Gamificação.

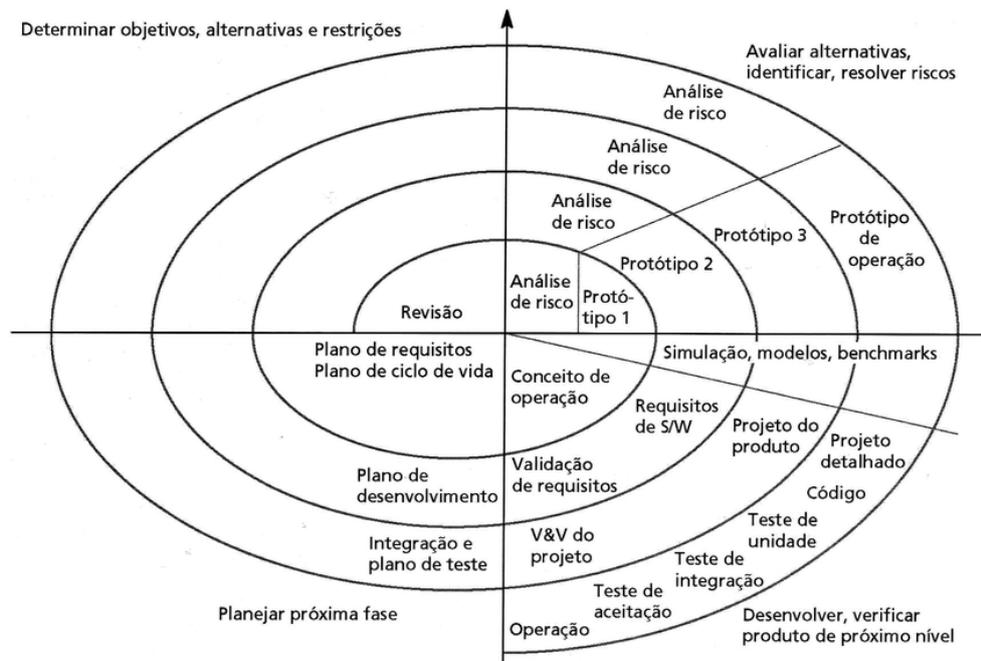
2.1 GERENCIAMENTO DE RISCOS EM PROJETOS DE SOFTWARE

No princípio, riscos eram uma preocupação, entretanto seu tratamento se dava através de estratégias reativas, que foram nomeadas por Tom Gilb: ‘escola Indiana Jones’, uma vez que na série de filmes, o personagem principal, ao enfrentar alguma adversidade, sempre dizia: “não se preocupe, vou pensar em alguma coisa! ”. Infelizmente ainda a maioria das equipes apoia-se ainda em uma estratégia reativa para o gerenciamento de riscos, de forma que ainda aguarda algum revés ocorrer no projeto para pensar em como esse revés será solucionado. (PRESSMAN e MAXIN, 2016)

Embora ainda haja algum debate sobre a melhor definição aplicável a risco, há certo consenso que o risco sempre está sempre vinculado a duas características principais: a incerteza e a perda. Isto é, o risco pode ou não ocorrer, não existe risco com probabilidade de 100%, nesse caso trata-se de uma restrição de projeto. E, em relação a perda, todo o risco, caso ocorra, impacta negativamente no projeto. (PRESSMAN e MAXIN, 2016)

Boehm, na década de 1990, foi pioneiro a incluir gerenciamento de risco no ciclo de vida de software, propondo o modelo em espiral. Há uma forte crítica ao modelo tradicional de desenvolvimento de software - cascata - e ao fato de que os desenvolvedores prometem implementar mais funcionalidades sem entender as implicações e os riscos associados a elas. (BOEHM, 1991)

Figura 1 - Modelo Espiral



Fonte: Boehm, 1991.

Na Figura 1 - Modelo Espiral está o diagrama do modelo espiral, que é um metamodelo cíclico e evolutivo que pode acomodar vários processos específicos, foi desenvolvido em base às experiências do autor com desenvolvimento no modelo cascata. Sua grande inovação tem sido orientar o desenvolvimento a partir da análise e planejamento de riscos, que são realizados durante todo o processo de desenvolvimento.

De acordo com Boehm (1991), identificar riscos e planejar ações de mitigação no início do projeto ajuda a reduzir custos e evitar impactos negativos no projeto. Sendo o pioneiro a explicitar a preocupação com riscos, após isto, diversas abordagens para o gerenciamento de riscos em projetos de software foram desenvolvidas.

Para Pressman e Maxin (2016), riscos podem ser subdivididos em riscos de projeto, riscos técnicos e riscos de negócio. Riscos de projeto, para os autores, identificam potenciais problemas de orçamento, cronograma, equipe, organização, recursos, clientes, complexidade do projeto e seu grau de incerteza, bem como requisitos; esses riscos estão diretamente ligados ao cronograma e ao custo do projeto. Já riscos técnicos englobam os relacionados a: projeto, implementação, interface, verificação e manutenção; riscos esses diretamente ligados à qualidade e ao prazo de entrega. Por fim os riscos de negócio têm impacto mais ligado a viabilidade do produto em si, sendo relacionados às seguintes áreas: mercado, não há real interesse no produto desenvolvido; estratégico, o produto desenvolvido não está de acordo

com a estratégia da empresa; vendas, o time de vendas não consegue/não sabe vender o produto; gerencial, perder o apoio da alta gerência devido a mudanças; ou, por fim, orçamentário, ter comprometimento dos profissionais envolvidos ou do orçamento.

O Software Engineering Institute – SEI – (PRESSMAN E MAXIN, 2016) identifica sete (07) princípios para a gestão de riscos em projetos de software. São eles:

- I. Perspectiva Global: os riscos devem ser encarados considerando não só o projeto, mas de tudo o que circunda, que constitui um contexto global;
- II. Visão antecipada: é fundamental estabelecer para os eventos futuros que sejam controláveis;
- III. Comunicação aberta: todos os membros da equipe devem ter liberdade para comunicar riscos em potencial a qualquer momento;
- IV. Integração: a gestão de riscos deve ser integrada a gestão de qualidade;
- V. Processo contínuo: o processo de gestão de riscos, integrado a gestão de qualidade, é um processo contínuo durante todo o desenvolvimento de software, e dessa forma a equipe precisa estar vigilante para atualizar os riscos conforme o processo caminha;
- VI. Visão Compartilhada do Produto: todos os envolvidos no processo precisam compartilhar da mesma visão do produto, de forma a identificar e analisar riscos de forma melhor;
- VII. Trabalho em equipe: os conhecimentos, as habilidades e os talentos de todos os envolvidos devem ser envolvidos nas atividades do processo de gestão de riscos.

O *Project Management Institute*, em seu Guia de Conhecimento de Gerenciamento de Projetos, propõe vários processos divididos em nove áreas de conhecimento. Toda uma área do conhecimento é voltada para o gerenciamento de riscos. Esta área de conhecimento inclui processos relacionados ao planejamento de gerenciamento de riscos, identificação, análise, planejamento de resposta, controle e monitoramento de riscos em um determinado projeto. Esses processos interagem entre si e com processos em outras áreas do conhecimento. (PMI, 2017)

A meta descrita pelo documento é reduzir a probabilidade e o impacto de eventos negativos e aumentar a probabilidade e o impacto de eventos positivos. O gerenciamento de risco aqui proposto consiste em um planejamento inicial, uma avaliação qualitativa, outra avaliação quantitativa, planejamento de resposta e monitoramento e controle.

De acordo com o corpo de conhecimento de gerenciamento de projetos (PMI, 2017), o gerenciamento de riscos contém os seguintes processos: planejar o gerenciamento de riscos, identificar riscos, realizar análises qualitativas de riscos, executar análises quantitativas de riscos, planejar respostas a riscos e monitorar riscos.

O primeiro processo implica em definir como serão conduzidas as atividades do gerenciamento de riscos para todo o projeto, definindo assim, como será o comportamento durante todas os processos subsequentes. São documentos considerados como entradas para esse processo o termo de abertura do projeto, o plano de gerenciamento do projeto, os documentos do projeto, os fatores ambientais da organização e os ativos de processos organizacionais. As principais ferramentas úteis são reuniões e análise dos dados, gerando-se assim o plano de gerenciamento de riscos.

Após isso, procede-se a identificação dos riscos, esse processo tem por objetivo determinar quais riscos podem afetar o projeto, bem como documentar suas características. Para tanto são consideradas as seguintes entradas: plano de gerenciamento de projeto, acordos, documentos de projeto, documentação de aquisições, fatores ambientais da organização, ativos de processos organizacionais e o plano de gerenciamento de riscos. Diversas ferramentas de processo podem ser utilizadas, as principais são coleta e análise de dados e listas de alerta (lista de categorias de risco pré-determinada, que serve para direcionar a identificação). Ao término do processo serão gerados os seguintes documentos: registro de riscos, relatório de riscos, bem como serão atualizados os documentos do projeto.

A análise qualitativa é a primeira etapa para a priorização, nela é avaliada a exposição ao risco a fim de priorizar quais riscos se tornarão objeto de análise e de ações adicionais. Dessa forma, são fundamentais a esse processo: o registro de risco, o plano de gerenciamento de risco, a linha de base do escopo, os fatores ambientais da organização e os ativos de processos organizacionais. Nessa etapa, é definida a categoria do risco com base na lista de alerta, ou na otimização de esforços, principalmente relacionados aos planos de resposta. Ao término se conduzirá a atualização dos documentos do projeto.

Análise quantitativa tem como objetivo analisar numericamente os riscos priorizados na análise qualitativa, pois é uma atividade de alta complexidade. São consideradas entradas a esse processo: plano de gerenciamento dos riscos, plano de gerenciamento de custos, plano de gerenciamento de cronograma, registros de riscos, fatores ambientais da organização e ativos de processo organizacionais. As principais ferramentas aqui são técnicas de modelagem e análise quantitativa de riscos, alguns exemplos dessas técnicas são: análise de sensibilidade,

análise de valor monetário esperado e Simulação Monte Carlo. Novamente ao término do processo, deverão ser atualizados os documentos do projeto.

O planejamento das respostas aos riscos tem por objetivo desenvolver opções para aumentar as oportunidades e diminuir as ameaças aos objetivos do projeto, dessa forma, nesse processo desenvolve-se respostas aos riscos priorizados, bem como é definido um proprietário do risco. As respostas aqui desenvolvidas implicam em mais recursos e atividades no orçamento, cronograma e plano de gerenciamento do projeto. São entradas o plano de gerenciamento do projeto, os documentos do projeto – principalmente o plano de gerenciamento dos riscos e o registro dos riscos, os fatores ambientais da empresa e os ativos do processo organizacionais. As principais estratégias para riscos são: prevenir, mitigar ou transferir. São saídas desse processo: solicitações de mudança, atualização do plano de gerenciamento de projetos e atualizações dos documentos de projeto.

A sexta edição do PMBoK trouxe o processo de implementação de respostas de forma explícita, anteriormente, essas atividades eram desenvolvidas no processo: Orientar e gerenciar o trabalho do projeto. Esse processo tem por objetivo implementar as respostas planejadas no processo: Planejar as respostas aos riscos. São entradas desse processo plano de gerenciamento do projeto, os documentos de projeto e os fatores ambientais da empresa. As ferramentas aqui variam de acordo com os riscos, com o contexto do projeto entre outros. Solicitações de mudanças e a atualização dos documentos do projeto são as saídas desse processo.

O último processo, monitor riscos, tem por objetivo controlar e monitorar os riscos de modo a: acompanhar os riscos identificados, monitorar a implementação dos riscos e respostas aos riscos, monitorar riscos residuais, identificar novos riscos, avaliar a eficácia do processo de riscos durante o ciclo de vida do projeto. As entradas do processo são o plano de gerenciamento do projeto, documentos de projeto, dados de desempenho do trabalho e relatórios de desempenho do trabalho. As ferramentas mais utilizadas nesse processo são análise de dados, reuniões e auditorias. As saídas do processo são informações sobre o desempenho do trabalho, solicitações de mudança, atualizações do plano de gerenciamento do projeto, atualizações de documentos do projeto e atualizações de ativos de processos organizacionais.

Além do PMBoK, outro documento que propõe um processo de gerenciamento de riscos é a NBR/ISO 31000. A norma organiza um processo de gerenciamento de riscos (ABNT, 2009), no qual são sete etapas básicas, descrição do contexto, identificação de riscos, análise de risco, avaliação de riscos, tratamento de riscos, monitoramento e análise crítica.

análise e comunicação. A comunicação é cumprida durante todas as fases. Para isso, propomos o uso de um mapa que comunica facilmente o status do tratamento dos riscos em qualquer momento. A descrição do contexto apresenta a compreensão do contexto externo, o que é importante para garantir que as metas e preocupações dos *stakeholders* sejam consideradas para desenvolver os critérios de risco.

Identificar as fontes de risco, áreas de impacto, eventos, suas causas e consequências potenciais para gerar uma lista abrangente de riscos com base nesses eventos que podem criar, aumentar, prevenir, reduzir, acelerar ou atrasar o alcance dos objetivos. A análise de risco envolve o desenvolvimento da compreensão dos riscos. A análise de risco fornece dados para a avaliação de risco e para decisões sobre a necessidade de tratar os riscos e sobre as estratégias e métodos mais apropriados para o gerenciamento de riscos. A análise de risco também pode fornecer subsídios para a tomada de decisões em que as escolhas precisam ser feitas e as opções envolvem diferentes tipos e níveis de riscos (ABNT, 2009). A avaliação de risco visa comparar os riscos identificados e os objetivos da organização, decidindo quais riscos valem a pena tratar. E as últimas fases de tratamento e monitoramento e controle são as atividades práticas que são realizadas para se preparar ou enfrentar um risco.

O Guia ISO 73/2009 define que os processos de gerenciamento de risco são o processo de aplicação sistemática de políticas, procedimentos e práticas de gestão para as atividades de comunicação, consultoria e estabelecimento do contexto e identificação, análise, avaliação, tratamento, monitoramento e risco. Juntamente com isso, define em seus 11 parágrafos termos relacionados à gestão de riscos.

O principal objetivo do documento é "incentivar uma compreensão consistente de uma abordagem coerente para a descrição de atividades relacionadas ao gerenciamento de riscos e o uso de terminologia uniforme de gerenciamento de riscos em processos e estruturas que lidam com o gerenciamento de riscos" (LUKO, 2013).

Embora haja várias tentativas de uniformizar as terminologias e os processos de gerenciamento de riscos, há uma infinidade de metodologias e práticas propostas para mitigar ativamente os riscos no desenvolvimento de software. Assim, o gerenciamento de riscos pode ser muito complexo e difícil de compreender e executar durante a produção de um software. Na prática, a gestão de riscos em projetos de desenvolvimento de software ainda é muitas vezes negligenciada e uma das razões é a falta de conhecimento dos fatores de risco que promovem uma baixa percepção deles no ambiente. (MENEZES JUNIOR et al, 2018)

2.3 GAMIFICAÇÃO

Gamificação é a aplicação de elementos de jogos a contextos diferentes dos de jogos. Isto é, a aplicação das teorias de diversão, como a apresentada por Johan Huizinga, que se debruçou sobre a questão da produção cultural humana, e que conclui que os jogos permeiam todas as fases da vida humana, a ponto de definir a própria produção cultural humana; a outros contextos. Gamificação vem sendo estudada há algum tempo, aparecendo em sites de vendas inicialmente, em que os usuários receberam pontos e medalhas por suas ações, dessa forma sendo incentivados a gastarem mais. (DUBOIS e TAMBURRELLI, 2013)

Desde os trabalhos de Huizinga, sabe-se que jogos estão ligados a natureza humana, mas, apenas em 1973, Charles Conradt indicou que as técnicas utilizadas em jogos para motivação poderiam ser portadas para outros ambientes. O termo gamificação foi cunhado pesquisador britânico Nick Pelling em 2003. Seguindo a ideia de que não é necessário retirar as pessoas de sua própria realidade para motivá-las, mas sim aplicar elementos que façam a realidade mais aprazível e divertida. O termo se popularizou mesmo a partir de 2008 como afirma Deterding *et al.* (2011), que em seu artigo busca uma definição real do termo gamificação, diferenciando-o de outros termos similares como jogos sérios.

A partir dessas primeiras experiências, levantou-se sua viabilidade como recurso educacional, o que transformou gamificação em uma tendência. Estudos sobre a aplicação da técnica nas áreas educacionais tornaram-se numerosos, e a partir disto, passou-se a ter consciência da possibilidade de aplicação dessa técnica em outras áreas. Kumar (2013) identifica algumas razões pelas quais seu uso tem sido profícuo nessas áreas, mas principalmente, aponta a geração digital que cresceu jogando vídeo games, utiliza nativamente mídias digitais e tecnologias móveis. Entretanto, em uma revisão sistemática de estudos, realizado por Pedreira *et al.* (2015) reforçado pelas evidências do trabalho desenvolvido por Belinazo e Fontoura (2017), os trabalhos que aplicam a técnica para engenharia de software são bastante iniciais.

Kumar (2013) afirma que diversas vezes se toma a abordagem que a autora chama de “brócolis coberto de chocolate”, uma vez que simplesmente se adicionam mecanismos como pontos, emblemas ou quadros de pontuação e chama-se a aplicação de gamificada. E não se pensa em uma nova solução em que sejam tratados os objetivos da gamificação. O desenvolvimento, em geral, tende a observar as necessidades do usuário, fazendo que o sistema seja eficaz e eficiente, no entanto, para o desenvolvimento de produtos gamificados

faz-se necessário ir além do conceito de usuário, e que se busque o conceito de jogador, uma vez que além da eficácia e eficiência, para um jogador, são necessários à diversão e ao prazer.

Para além desses aspectos, é fundamental que a gamificação consiga desenvolver regras de jogo significativas e envolventes, ao mesmo tempo, integradas às regras de negócio. Apresentar uma interface atraente, que seja conectada a interface de usuário séria. Bem como, se põe como desafio avaliar a gamificação em sua usabilidade e em seu poder de retenção do usuário. (Ašeriškis e Damaševičius, 2014)

Pesquisas apresentadas por Ašeriškis e Damaševičius (2014), relatam que de forma geral o uso da gamificação deve ser ampliado nos próximos anos, isto é, 53% das pessoas pesquisadas pelo Pew Research Center expuseram que, até 2020, o uso da gamificação será generalizado. Um estudo bem conhecido do Instituto Gartner afirmou que, até 2015, mais de 50% das organizações que gerenciam processos de inovação irão gamificar tais processos. Mais de 70% das empresas da Forbes Global 2000 planejam usar pelo menos alguns elementos de gamification para marketing de produtos e retenção de clientes.

Ao longo dos anos os estudos em gamificação maturaram conforme afirma Nacke & Deterding (2017), e tornaram-se um campo multidisciplinar, enquanto a maioria dos estudos iniciais apresentavam falta de um arcabouço teórico, rigor metodológico e inovação propriamente dita, isso tem mudado ao longo dos anos. Nesses termos, os estudos têm testado isoladamente os elementos do design gamificado.

Segundo os autores (Nack & Deterding, 2017), esse é um importante primeiro passo, embora o uso de pontos, distintivos, níveis e placares tenha sido amplamente explorado nesses estudos, ainda há um grande vazio na exploração de cânones das teorias de jogos quando se pensa em direção a objetivos, teoria da autodeterminação e etc. os autores ainda apontam que não há estudos rigorosos que utilizem-se de ambos mediadores psicológicos e saídas comportamentais.

Dessa forma, para que se possa compreender melhor os processo motivacionais faz se necessário um suporte psicológico sobre a motivação e a motivação no mundo do trabalho. Entende-se na Psicologia, a motivação quanto processo psicológico básico capaz de fornecer impulsos para a realização de determinadas tarefas, sendo essas de natureza biológica ou por demandas aprendidas imerso em um tecido social, a relevância de abordar a motivação faz-se necessário para que possamos compreender, intervir e reelaborar estratégias para efetivarmos o prazer em executar um comportamento para alcançarmos um objetivo, conforme Catania (1979): “mudanças na significância de estímulos são a preocupação básica do estudo da motivação”. Compreender aspectos de produção de subjetividade relacionados aos processos

de trabalho nas organizações facilita o desenvolvimento de fontes de prazer que o trabalhador encontra ou pode encontrar no seu ambiente de produção, trazendo questões como conforto, familiaridade com as ferramentas técnico operacionais sob uma perspectiva contraposta a teoria da administração científica de Taylor, onde o salário era o motivador principal.

“A motivação pode ser entendida ora como fator psicológico, ou conjunto de fatores, ora como um processo. Existe um consenso entre os autores quanto a dinâmica desses fatores psicológicos ou do processo, em qualquer atividade humana. Eles levam a uma escolha, instigam, fazem iniciar um comportamento direcionado a um objetivo”. (Bzuneck, 2004).

Salientando as relações indivíduo e trabalho, é importante destacarmos que cada indivíduo apresenta um perfil motivacional, um motivo para a ação, essa se estabelece de forma motivacional valorativa, forma de vontade (Nietzsche, 1970) relação de pertencimento ao ambiente de trabalho, entre questões culturais e sociais, contudo há estruturas motivacionais comum a todos, como o prestígio.

Uma análise aos aspectos teóricos de motivação no trabalho é notório as questões de base que servem de suporte para a compreensão das teorias motivacionais no trabalho e conseguinte a leitura desse processo cognitivo, enquanto a administração científica transfere ao capital a principal forma de motivação, desconsiderando a dinâmica entre indivíduo e a execução do seu trabalho, podemos ressaltar a formulação de novos campos conceituais referentes a motivação, dando significação a interface indivíduo e ambiente, chamado de aspectos motivadores do trabalho, estes desempenham importante papel na promoção de bem estar subjetivo e potencialização das práticas de trabalho, conhecida como teoria exógena denominada por Katzell e Thompson (1990). Portanto, conhecer os processos motivacionais é poder compreender o que nos move ao trabalho e suas implicações, seja no conteúdo, ambiente ou quanto intervenção, com o propósito de modificar formas de trabalho pré-estabelecidas com um baixo índice de produtividade e satisfação, tornando estas atrativas com o intuito de estimular a funcionalidade do trabalho.

3. TRABALHOS RELACIONADOS

O estudo de Schönbohm, A. e Jülich, A. (2016) analisa os benefícios potenciais da gamificação em riscos para pequenas e médias empresas, o estudo utiliza uma abordagem de pesquisa-ação, na qual aplica-se um método teórico e outro semiexperimental, em que os sujeitos da pesquisa participaram de oficinas sobre o gerenciamento de riscos de maneira gamificada. O estudo aponta para uma maior motivação dos participantes, um aprendizado rápido, sugerindo assim que a gamificação é uma ferramenta valiosa para o gerenciamento de riscos, em um contexto geral.

Um mapeamento sistemático sobre gamificação em engenharia de software foi conduzido por PEDREIRA *et al.* (2015), estudo esse que apontou que a gamificação na área encontra-se ainda em um estágio bastante inicial. O estudo considerou publicações de janeiro de 2011 a junho de 2014, estudos que em sua maioria apresentaram o desenvolvimento de software como foco da inserção de elementos de jogos, principalmente nas atividades em que o cliente era envolvido. Há um grande número de estudos que apresenta apenas a introdução de mecanismos rudimentares de gamificação, como um sistema de pontuação e emblemas, e desses estudos poucos fornecem evidências claras dos impactos positivos da aplicação da gamificação.

Em 2017, conduziu-se uma revisão sistemática de forma a conhecer o estado da arte da aplicação de gamificação em engenharia de software (BELINAZO E FONTOURA, 2017). Os resultados obtidos durante a análise dos estudos existentes sobre gamificação foi que a maioria ainda é bastante preliminar, oferecendo pouca evidência empírica sólida sobre o engajamento, motivação e o desempenho de suas propostas. Além de apenas adicionarem mecânicas de jogos a processos e sistemas complexos, não criando um projeto centrado sobre a gamificação, de forma a corroborar os resultados de Pedreira *et al.* (2015). A partir do estudo desenvolvido em 2017 por Belinazo e Fontoura foram selecionados os estudos apresentados a seguir como trabalhos relacionados, por serem os mais próximos a temática do trabalho.

Garcia *et al.* (2017) propuseram uma estrutura para a aplicação de gamificação em engenharia de software, baseada no uso de uma ontologia, uma metodologia e um suporte tecnológico. A ontologia apresenta um metamodelo, no qual são apresentados os conceitos mais relevantes para gamificação em engenharia de software. Como suporte é apresentada uma metodologia que auxilia no processo de gamificação definindo as atividades *gamificadas*. Por fim, é proposta uma API de integração de ferramentas, que gera um *website*

dimensionado a partir das informações coletadas das ferramentas de casos já utilizadas pela equipe.

Medeiros *et al.* (2015) propôs a gamificação do processo de software a partir do *framework SCRUM*, utilizando princípios de Role Playing Game - RPG, a fim de maximizar o resultado esperado. A gamificação é alcançada por meio do mapeamento dos objetivos do *SCRUM* para os desafios do jogo. Eles são definidos pelos autores, a partir de métricas de qualidade, representadas por meio de conquistas. No entanto, o estudo em geral aponta que os sujeitos optaram pela velocidade de produção, abdicando da qualidade do produto gerado ao utilizar este método.

Com isso em mente, o estudo de Schönbohm, A. e Jülich, A. (2016) propôs o uso de técnicas de gamificação para o gerenciamento de riscos, entretanto não apresenta uma metodologia para sua aplicação. O estudo, validado por meio de uma pesquisa-ação, está em um estágio muito inicial, e ainda é sobre riscos para projetos de uso geral, não sendo explicitamente aplicado ao desenvolvimento de software. Garcia *et al.* (2017) propõe a aplicação de técnicas de gamificação, criando um dicionário geral, em engenharia de software, mas sua abordagem é bastante genérica e, ao que parece, apenas adiciona elementos de gamificação ao processo, em vez de pensar em um processo completamente gamificado. E, finalmente, Medeiros *et al.* (2015) propõem um sistema escalável para o gerenciamento do *SCRUM*, mas as métricas de qualidade do processo não são o foco, e o tratamento de risco não é enfoque dessa abordagem. Ao passo que a proposta de abordagem gamificada aqui apresentada se debruça sobre o gerenciamento de riscos em projetos de software, bem como apresenta a sua estrutura baseada em um jogo específico, dando ao seu usuário diretrizes de como ser construída, bem como apresenta uma ampliação nos controles dos processos, e pode ser utilizada como apoio para metodologias ágeis e planejadas.

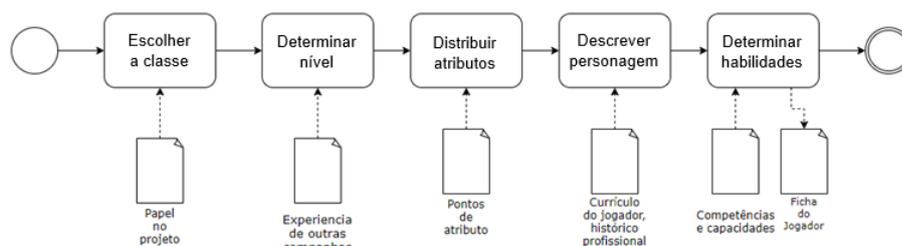
4. UMA ABORDAGEM GAMIFICADA PARA O GERENCIAMENTO DE PROJETOS

Essa abordagem para o gerenciamento de projetos de software desenvolvida no Laboratório de Computação Aplicada da Universidade Federal de Santa Maria. Para o desenvolvimento desta abordagem, selecionou-se como metáfora fundamental jogos de *Role Playing Game* - RPG, uma vez que esse gênero de jogo apresenta uma estrutura que pode apoiar a gestão. A estrutura de um jogo RPG é composta por: jogadores, história, *dungeon master*, monstros, prêmios, todos esses abrangidos por um mapa, que representa a aventura de campanha. A solução proposta apresenta uma metodologia completa para gerenciar projetos de desenvolvimento de software baseados no jogo *Dungeons & Dragons (D&D)*. D&D é um RPG, estruturado e imersivo, em que um grupo de jogadores controla personagens criados por eles, em uma aventura estabelecida e gerenciada pelo *dungeon master* (Souza, 2019 no prelo).

Para a construção da abordagem foi elaborado um paralelo entre elementos do jogo D&D e elementos da Engenharia de Software. O *dungeon master*, o mestre, organizador e criador dos detalhes da campanha, associa-se ao gerente da equipe, que pode ser nomeado de forma diferente dependendo do processo adotado. Os jogadores são os demais membros da equipe envolvida no projeto. A ficha de personagem foi adaptada para que os atributos, habilidades e descrições detalhassem as habilidades dos membros da equipe. O mundo do jogo é composto pela empresa e o projeto contextualizado. Campanha, em D&D são arcos de *storytelling*, na abordagem uma campanha pode ser considerada o projeto como um todo, ou etapas bem definidas de um projeto. Aventuras foram associadas às iterações de desenvolvimento.

Para que o jogo se inicie, é preciso que o jogador em primeira instância complete seu perfil, e para tanto ele precisará completar as atividades mostradas na Figura 2.

Figura 2 – Processo de criação do personagem



Fonte: Souza, 2019 – no prelo.

Após a execução das cinco etapas exibidas na Figura 2, será gerada a ficha do personagem, que resume o perfil profissional, que balizará a escolha das tarefas que serão delegadas ao profissional. Na **Figura 3** pode ser visto o produto final do processo.

Figura 3 - Ficha de personagem

The screenshot displays the 'Ficha de personagem' (Character Sheet) for Gabriel Belinazo. The interface is organized into several sections:

- Header:** 'Ranger' logo and a 'Sair' (Logout) button.
- Navigation Sidebar (MESTRE/JOGADOR/ACCOUNT SETTINGS):**
 - MESTRE:** Campanhas, Aventuras, Missões, Regras, Recompensas, Monstros, Classes, Riscos, Habilidades.
 - JOGADOR:** Ficha de personagem (selected).
 - ACCOUNT SETTINGS:** Profile, Change Password.
- Main Content Area:**
 - Character Name:** Gabriel Belinazo.
 - Class:** Analista de Sistemas.
 - Level:** 6 Junior I.
 - Experience:** 14000 (próximo nível 23000).
 - Attributes:**
 - LÓGICA: 2
 - AGILIDADE: 1
 - CREATIVIDADE: 2
 - AÇÃO: 2
 - COMUNICAÇÃO: 5
 - RESILIÊNCIA: 3
 - Habilidades (Skills):** Banco de dados, Negociação. Includes a 'Selecionar' dropdown and a '+ ' button.
 - Prêmios (Rewards):** Empty section.
 - Descrição:** Empty text area.
- Right Panel:**
 - Character Name:** Gabriel Belinazo.
 - Buttons:** Campanha, Histórico.
 - Section:** Campanhas.
 - Buttons:** Campanha, Histórico.

Fonte: Autor.

A ficha do personagem deverá ser atualizada continuamente durante o desenvolvimento das aventuras e campanhas que o jogador participará. Na **Figura 3** pode ser visualizada uma ficha preenchida com os dados de um jogador fictício. Nesse exemplo, o jogador selecionou a classe 'Analista de sistemas', uma vez que sua principal função no projeto é a análise geral do projeto. Após isso, iniciou, por padrão, no nível um e avançará seu nível adquirindo pontos de experiência por completar tarefas. Por fim, distribuiu seus pontos iniciais em atributos.

O sistema gamificado é composto por um conjunto de atividades: criar personagens; descrever projetos e seu contexto; definir o processo de software, requisitos e desafios do projeto; estabelecer regras de negócio, requisitos e práticas não funcionais a serem utilizadas no projeto. Nesse aspecto, *a priori* das configurações de usuário, é necessário que sejam configuradas.

O jogo se subdivide em campanhas, as campanhas, que representam o projeto, são constituídas de um conjunto de aventuras. As aventuras, que representam as iterações do projeto, são compostas de tarefas, que são as atividades reais do projeto. As tarefas têm uma descrição e uma lista de habilidades e atributos necessários para a sua execução. Um jogador pode escolher tarefas a serem executadas por ele na aventura, com base em seus atributos e

habilidades descritos em sua ficha. O *dungeon master* pode atualizar a configuração do jogo, desafios, restrições e regras, ou atribuir jogadores a tarefas durante o jogo.

As aventuras têm um tempo de execução associado a elas e seus objetivos são comuns à equipe. Durante a aventura, os jogadores podem escrever um diário, um registro dos eventos que aconteceram na aventura pela perspectiva do jogador. No final da aventura, o *dungeon master* pode distribuir recompensas, como conquistas e pontos de habilidade, para os jogadores enquanto eles executam suas tarefas.

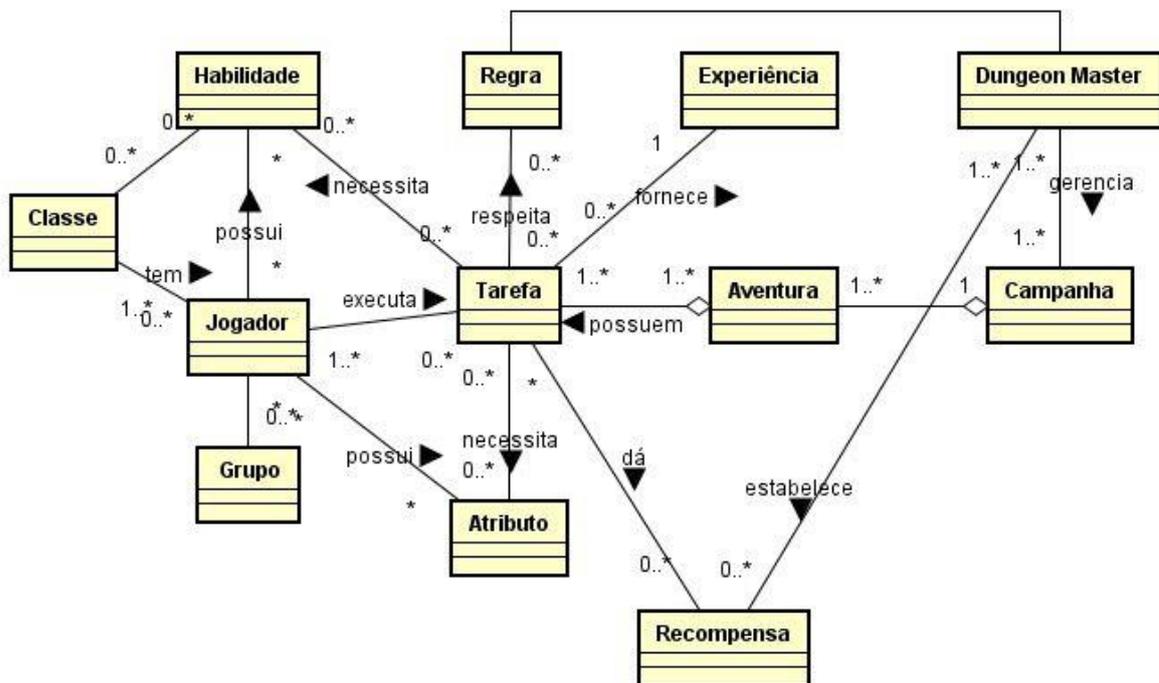
Durante as campanhas, o DM gerenciará no sistema gamificado, inserindo, alterando aventuras e o mundo. Os personagens podem interagir entre si, de modo a aumentar a comunicação da equipe. Ao final da campanha, ocorre o encerramento e consolidação do histórico da campanha, nessas etapas, o DM escolherá quais recompensas serão mantidas, e comporão o mundo e as próximas aventuras. O encerramento possibilita a avaliação da evolução das pessoas, bem como uma retrospectiva completa do projeto de software.

5. A ABORDAGEM GAMIFICADA PARA O GERENCIAMENTO DE RISCOS

A abordagem gamificada de gerenciamento de projetos de software para processos planejados e processos ágeis é discutida em trabalho desenvolvido no Grupo de Pesquisa em Engenharia de Software (PEEnSo), do Laboratório de Computação Aplicada (LaCA) da Universidade Federal de Santa Maria (UFSM) no qual essa dissertação está inserida. Assim, essa dissertação tem como escopo a seguinte pergunta: Como gerenciar riscos em projetos de software com processos adaptados às características da equipe e do projeto, promovendo a maturidade dos processos de software, mas mantendo a equipe comprometida e motivada?

O metamodelo a seguir, desenvolvido por Souza (2019, no prelo), apresenta os conceitos envolvidos na abordagem de gerenciamento de projetos gamificada, como jogador, *dungeon master*, tarefas, atributos e habilidades.

Figura 4 – Metamodelo da abordagem gamificada de gerenciamento de projetos



Fonte: Souza, 2019 – no prelo.

Para a construção dessa abordagem considerou-se uma associação dos elementos descritos no metamodelo, exibido na Figura 4 – Metamodelo da abordagem gamificada de gerenciamento de projetos, a elementos da disciplina de Engenharia de Software,

considerando as características e as responsabilidades das disciplinas e suas possíveis correlações com mecânicas de jogo. Por exemplo, classe de jogador é um conjunto de atributos e habilidades que se espera do jogador de forma a relacionar-se diretamente com o papel desempenhado pelo jogador no projeto. Joga-se pela execução de campanhas que são formadas por aventuras. As aventuras consistem em conjuntos de tarefas, que são atividades reais do processo de software.

Cada tarefa apresenta sua própria descrição real e lúdica, bem como uma lista de habilidades, atributos e restrições. As habilidades, os atributos e as restrições da tarefa servem para selecionar quais são os jogadores que idealmente devem desenvolver a tarefa. Aventuras representam uma iteração. Podendo representar marco, conjunto de requisitos, ou etapa bem definida do projeto. Entretanto, ao fim das aventuras, deve-se ter o produto de software pronto.

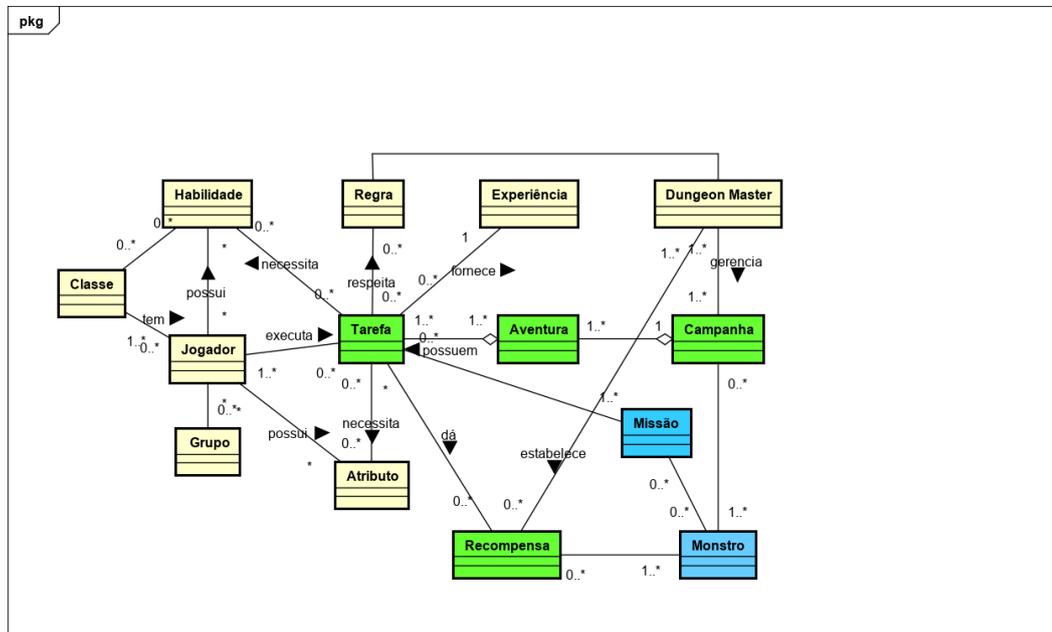
Entretanto, a abordagem, como posta por Souza (2019 – no prelo) não prevê o tratamento de riscos nem a melhoria de processo. Portanto, propõe-se aqui uma abordagem gamificada para o gerenciamento de riscos em processos de software, bem como itens para a melhoria contínua do processo de software.

De forma a estruturar as principais atividades e sequência de ações que descrevem a abordagem temos o metamodelo ampliado, pela inclusão das classes relacionadas ao gerenciamento de riscos, que estão realçadas em azul no modelo exibido na Figura 5.

Tendo em vista as seguintes fases do ciclo de vida de gerenciamento de riscos: 1. Identificação de riscos; 2. Análise qualitativa de risco; 3. Seleção das práticas a serem aplicadas no contexto do projeto; 4. Monitorar as práticas através do jogo; 5. Avaliação de práticas e lições aprendidas durante o projeto. Para realizar todos esses passos, o sistema gamificado proposto por Souza (2019 - no prelo) foi expandido de acordo com o metamodelo apresentado na Figura 5 - Metamodelo da abordagem de gerenciamento de riscos gamificado.

Na Figura 5 - Metamodelo da abordagem de gerenciamento de riscos gamificado, pode-se observar que o sistema de gerenciamento de risco gamificado traz os conceitos de jogador, grupo de jogadores, missão, monstro e recompensas. Para o jogador, são usados atributos e habilidades. A missão consiste em prática destinada a evitar um risco, mitigar seus danos ou mesmo a preparação para enfrentá-lo. O risco é mapeado para um monstro que pode ou não ser encontrado durante a campanha. A campanha é a iteração no processo de desenvolvimento de software. Dessa forma, a cada iteração, se deve passar por todas as fases do ciclo de vida do gerenciamento de riscos, como no modelo espiral, proposto por Boehm (1991).

Figura 5 - Metamodelo da abordagem de gerenciamento de riscos gamificado



Fonte: Autor.

No diagrama são representados em amarelo os elementos da abordagem *Ranger*, que foram definidos por Souza (2019, no prelo). Em verde, estão destacados os elementos da abordagem que foram alterados, e em azul os adicionados a essa abordagem. A tarefa são as atividades do processo, que ao serem executadas geram recompensas ao jogador, entretanto, ao ligar uma tarefa a um monstro criamos uma missão. O conjunto de missões e o monstro representa o plano de risco. No caso de o DM entender que o plano de gerenciamento de riscos falhou, isto é, o risco se efetivar, o monstro terá como efeito a reorganização das tarefas e da aventura.

Para a implementação do sistema gamificado foram definidos as seguintes histórias de usuários:

US024 - Selecionar monstros

US025 - Criar monstros

US026 - Selecionar missões

US027 - Criar missões

US028 - Acompanhar a campanha

- US029 - Gerenciar a campanha
- US030 - Definir poder de monstro
- US031 - Sugerir monstro
- US032 - Segerir monstro
- US033 - Histórico de campanha
- US034 - Definição de threshold
- US035 - Finalização da campanha
- US036 - Atribuir missões

O detalhamento dessas histórias de usuários que descrevem os requisitos implementados na ferramenta Ranger para compatibilizá-la com o gerenciamento de riscos, bem como das demais histórias do sistema, desenvolvidas por Souza (2019 – no prelo) encontram-se no Anexo A.

O uso de monstros, como uma representação dos riscos do projeto de software, foi pensado a partir dos monstros descritivos do livro dos monstros do jogo *Dungeons and Dragons 5th Edition*, assim como outros elementos e mecânicas do *Role Playing Game*. Dessa forma, temos a classificação em termos de monstrosidade e alinhamento. Por monstrosidade consideramos o grau de impacto do risco no projeto, para que o prazo de risco, 'custo' e 'necessidade do cliente' sejam de monstrosidade titânica, ou seja, para o contexto não é possível derrotá-los. O encontro é como tratamos quando o risco se torna uma realidade no projeto. E para cada um desses riscos titânicos, temos uma habilidade especial que pode ser representativa das medidas tomadas para mitigar ou evitar esses encontros, que são a blindagem para questões relacionadas a custos, às necessidades do cliente e ao prazo.

O jogador possui uma série de atributos lúdicos e reais, tendo como currículo o perfil do jogo, pelo qual será possível ao sistema gamificado distribuir missões de acordo com o perfil. Armadura, resiliência e resistência são representações lúdicas das habilidades necessárias para mitigar os danos e assumir riscos negativos. O jogador pode criar uma história para seu personagem e atribuir pontos a seus atributos enquanto dirige seu perfil pela primeira vez, a fim de dar subsídios para a divisão de missões e posterior avaliação das métricas geradas. Monstros podem ser evitados, enfraquecidos ou enfrentados em um encontro.

Portanto, existem práticas e tarefas, que são tratadas como missões que devem ser realizadas para que os monstros possam ser derrotados ou enfraquecidos, ou a equipe esteja

equipada para o encontro. O conjunto de práticas e atividades é denominado campanha. Em D&D, os monstros são representados pela ficha exibida na Figura 6.

Figura 6 - Ficha de monstro em D&D



DRAGÃO NEGRO ADULTO						
<i>Dragão Enorme, caótico e mau</i>						
Classe de Armadura 19 (armadura natural)						
Pontos de Vida 195 (17d12 + 85)						
Deslocamento 12 m, natação 12 m, voo 24 m						
FOR	DES	CON	INT	SAB	CAR	
23 (+6)	14 (+2)	21 (+5)	14 (+2)	13 (+1)	17 (+3)	
Testes de Resistência Des +7, Con +10, Sab +6, Car +8						
Perícias Furtividade +7, Percepção +11						
Imunidade a Dano ácido						
Sentidos percepção às cegas 18 m, visão no escuro 36 m, Percepção passiva 21						
Idiomas Comum, Dracônico						
Nível de Desafio 14 (11.500 XP)						
Anfíbio. O dragão pode respirar ar e água.						
Resistência Lendária (3/Dia). Se o dragão falhar em um teste de resistência, ele pode escolher obter sucesso, no lugar.						
AÇÕES						
Ataques Múltiplos. O dragão pode usar sua Presença Aterradora, ele pode então, realizar três ataques: um com sua mordida e dois com suas garras.						
Mordida. <i>Ataque Corpo-a-Corpo com Arma:</i> +11 para atingir, alcance 3 m, um alvo. <i>Acerto:</i> 17 (2d10 + 6) de dano perfurante mais 4 (1d8) de dano de ácido.						
Garra. <i>Ataque Corpo-a-Corpo com Arma:</i> +11 para atingir, alcance 1,5 m, um alvo. <i>Acerto:</i> 13 (2d6 + 6) de dano cortante.						
Cauda. <i>Ataque Corpo-a-Corpo com Arma:</i> +11 para atingir, alcance 4,5 m, um alvo. <i>Acerto:</i> 15 (2d8 + 6) de dano de concussão.						
Presença Aterradora. Cada criatura, à escolha do dragão, que esteja a até 36 metros do dragão e esteja ciente disso, deve ser bem sucedida num teste de resistência de Sabedoria CD 16 ou ficará amedrontada por 1 minuto. Uma criatura pode repetir o teste de resistência no final de cada um dos turnos dela, terminando o efeito sobre si, caso obtenha sucesso. Se o teste de resistência de uma criatura for bem sucedido ou caso o efeito termine sobre ela, a criatura ficará imune a Presença Aterradora do dragão pelas próximas 24 horas.						
Sopro Ácido (Recarrega 5–6). O dragão pode expelir ácido em uma linha com 18 metros de comprimento e 1,5 metro de largura. Cada criatura na linha deve realizar um teste de resistência de Destreza CD 18, sofrendo 54 (12d8) de dano de ácido se falhar na resistência, ou metade desse dano caso obtenha sucesso.						
AÇÕES LENDÁRIAS						
O dragão pode realizar 3 ações lendárias, escolhidas dentre as opções abaixo. Apenas uma ação lendária pode ser usada por vez e apenas no final do turno de outra criatura. O dragão recupera as ações lendárias gastas no começo do turno dele.						
Detectar. O dragão realiza um teste de Sabedoria (Percepção).						
Ataque com Cauda. O dragão realiza um ataque de cauda.						
Ataque com Asas (Custa 2 Ações). O dragão bate suas asas. Cada criatura a até 3 metros do dragão deve ser bem sucedida num teste de resistência de Destreza CD 19 ou sofrerá 13 (2d6 + 6) de dano de concussão e cairá no chão. Após bater suas asas, o dragão pode voar até metade do seu deslocamento de voo.						

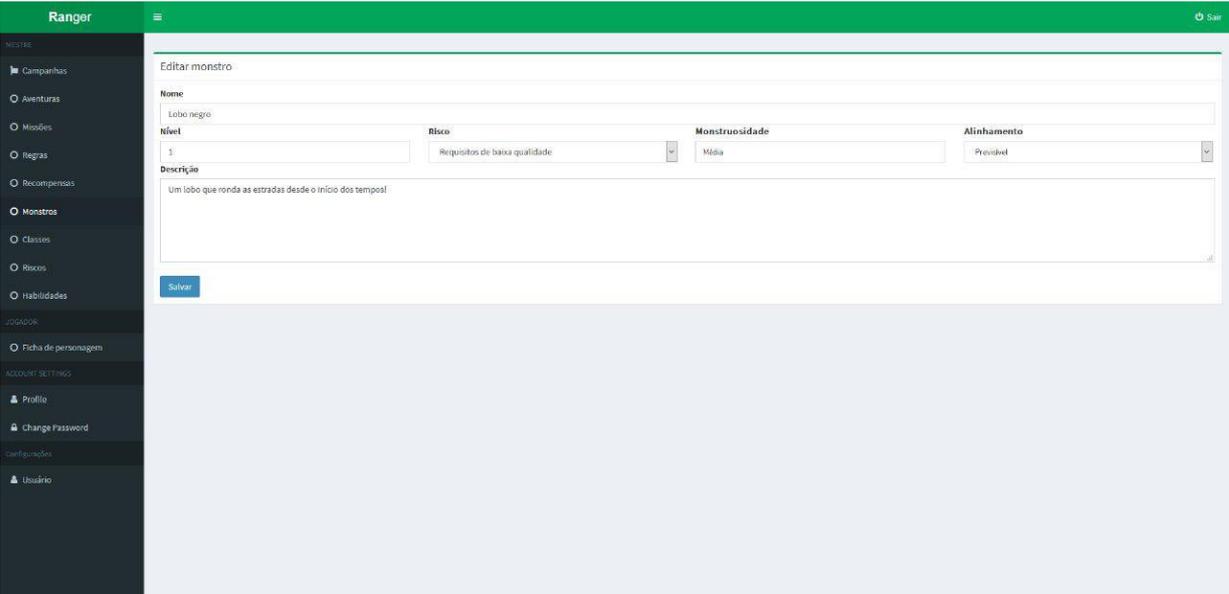
Fonte: Mearls, 2014.

Alinhamento, para o jogo, é uma maneira de entender o comportamento geral do monstro, consistindo de conjunto de possibilidades. Para o contexto, consideramos o alinhamento como sendo a natureza do risco, entendendo como leal o risco de predição fácil,

neutro intermediário e caótico o que não pode ser facilmente previsto. Assim, podemos organizar os monstros a serem selecionados para serem combatidos e suas campanhas podem ser mais bem organizadas. Os monstros têm níveis, que são dados pela probabilidade da matriz versus o grau de impacto do risco, para serem informados pelo gerente do projeto. De forma que um risco que tem alta probabilidade e alto grau de impacto deve apresentar uma campanha maior que aquela com baixa probabilidade e baixo grau de impacto.

O jogo sempre apresentará três monstros principais, que não são opcionais, que se tratam dos aspectos fundamentais de projetos: custo, prazo e necessidade do cliente. Esses monstros são considerados de monstruosidade titânica, que não podem ser derrotados em caso de um encontro, que é o não cumprimento do prazo do projeto, falha em atender ao custo acordado e não cumprimento das necessidades do cliente. Vê-se que a ocorrência de um destes implica na falha total ou parcial do projeto de software. Desta forma eles são representados por monstros que não podem ser derrotados no jogo, e são acompanhados por monstros menores que podem ser enfrentados, e desta forma, os três maiores riscos do projeto podem ser evitados.

Figura 7 - Ficha de monstro



A imagem mostra a interface de usuário 'Ranger' com um formulário para editar um monstro. O formulário contém os seguintes campos:

- Nome:** Lobo negro
- Nível:** 1
- Risco:** Requisitos de baixa qualidade
- Monstruosidade:** Média
- Alinhamento:** Previsível
- Descrição:** Um lobo que ronda as estradas desde o início dos tempos!

Um botão 'Salvar' está visível abaixo do campo de descrição.

Fonte: Autor.

Como pode ser visto na Figura 7 - Ficha de monstro, os monstros recebem um nome, neste caso “Lobo Negro”, eles recebem um alinhamento, que representa sua previsibilidade, no caso apresentado, significa neutro ter uma previsibilidade mediana. O lobo negro

representa o risco de “baixa qualidade”, e a breve descrição e ações do monstro servem para guiar a seleção de missões que podem ser usadas no jogo quando esse risco é selecionado. As missões que podem compor a campanha, considerando também o risco exposto na Figura 8 que apresenta o risco como foi cadastrado no sistema.

Figura 8 - Riscos

The screenshot shows the 'Editar risco' form in the Ranger application. The form is titled 'Editar risco' and contains the following fields and content:

- Nome:** A text input field.
- Requisitos de baixa qualidade:** A text input field.
- Nivel:** A text input field containing the value '1'.
- Descrição:** A text area containing the text 'Requisitos com baixa qualidade implicam em: retrabalho, mudança constante, não cumprimento de prazo.'
- Salvar:** A blue button at the bottom left of the form area.

Fonte: Autor

Na Figura 9 é mostrada a tela de cadastro de tarefas, as tarefas representam atividades reais do processo de produção de software. As tarefas ao serem relacionadas a um monstro tornam-se missões.

Figura 9 - Tela para adição de tarefas

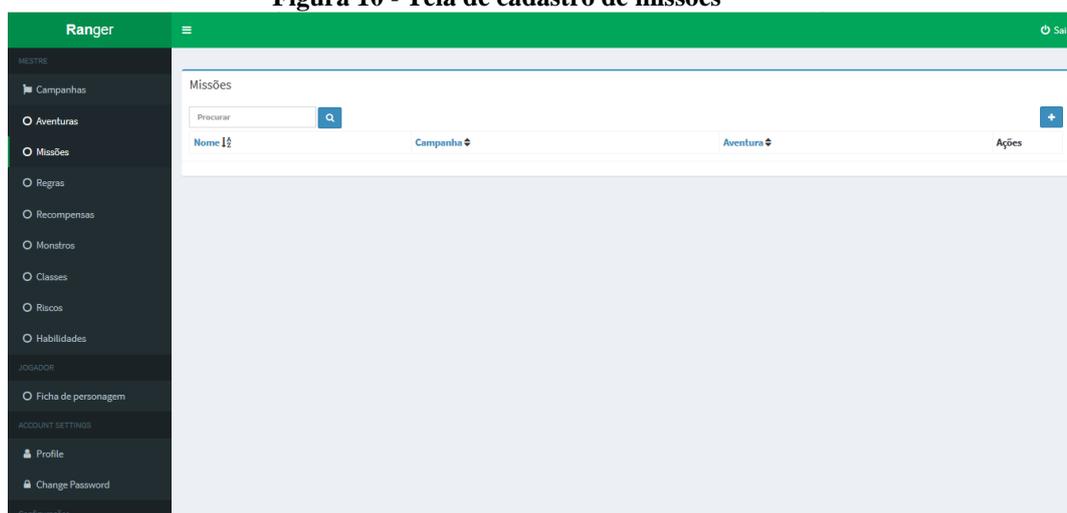
The screenshot shows the 'Adicionar Tarefa' form in the Ranger application. The form is titled 'Adicionar Tarefa' and contains the following fields and content:

- Nome:** A text input field.
- Prioridade:** A text input field.
- Status:** A text input field.
- Aventura:** A dropdown menu with 'Selecionar' selected.
- Aventura:** A second dropdown menu with 'Selecionar' selected.
- +Habilidades, +Atributos, +Regras, +Recompensas:** Four green buttons with plus signs and labels.
- Descrição:** A text area.
- Salvar:** A blue button at the bottom left of the form area.

Fonte: Autor.

A campanha consiste em missões, que traduzem as práticas comumente usadas para mitigação, ou para a execução do plano de tratamento de um risco. Na Figura 10 é mostrada a tela de cadastro de missões, isto é, ao se selecionar uma tarefa e vinculá-la a um monstro, transforma-se a tarefa em uma missão. Para isso a tela exibe uma opção para buscar missões cadastradas e ao selecioná-las elas ficam serão vinculadas a um monstro.

Figura 10 - Tela de cadastro de missões



Fonte: Autor

Além disso, na campanha pode haver nenhum encontro ou vários, pois um risco pode ser completamente evitado. Estes eventos serão selecionados pelo DM de acordo com as necessidades do projeto, e podem ser ajustados durante o jogo. A campanha é representada em um mapa que organiza as fases de desenvolvimento de software. Os jogadores e os riscos podem aparecer em pontos específicos no mapa, uma vez que determinados riscos ou certas ações para mitigação só podem ser desenvolvidas em estágios específicos do processo de software.

As configurações são de responsabilidade do gerente, o que aumenta o tempo de preparo do jogo, ao mesmo tempo em que o especifica para a equipe e o projeto, a fim de atender melhor as demandas do projeto. Além disso, no final de cada campanha, o sistema gamificado gera um relatório que traz métricas e estatísticas para que a equipe possa aprender mais e melhorar, aplicando a melhoria contínua e aprendendo com os erros, as práticas recomendadas e as lições aprendidas. Assim, o sistema pode ser aplicado na organização para organizar os fluxos de gerenciamento de risco para vários projetos.

Dessa forma, a abordagem adotada é uma abordagem qualitativa, em que os riscos são identificados, analisados, priorizados, é definido o plano do risco, os riscos são gerenciados e, durante todo o processo, os planos de risco são monitorados. Para a abordagem, essas fases são realizadas da seguinte maneira: a identificação de riscos se dá pela verificação de um *checklist* pré-cadastrado ou cadastrando novos. A análise de riscos é desenvolvida fora da ferramenta, porém segue a lógica da Tabela 1 - Relação entre D&D e a abordagem gamificada em que está explicitado como os atributos de riscos devem ser convertidos para suas versões gamificadas. Após, os riscos priorizados viram monstros na ferramenta, isto é, relaciona-se um risco e um monstro, e o plano do risco é criado

Tabela 1 - Relação entre D&D e a abordagem gamificada

<i>Atributo</i>	<i>Explicação</i>	<i>Abordagem Gamificada</i>
Tamanho	Define a dimensão de um monstro. Pode ser Miúdo, Pequeno, Médio, Grande, Enorme ou Imenso	Avaliação qualitativa do risco priorizado.
Tipo	Indica a natureza fundamental de um monstro.	Indica a natureza do projeto em que o risco atua diretamente, seja prazo, orçamento ou necessidade do cliente.
Tendência / Alinhamento	Oferece uma orientação relativa ao comportamento e orientação do monstro. Por exemplo, um monstro caótico e mau provavelmente será agressivo e impetuoso.	Natureza da predição.
Pontos de Vida	Determina quanto de vida um monstro possui. Um monstro geralmente fica inconsciente ao chegar a 0 pontos de vida.	-
Deslocamento	Estabelece quão longe um monstro pode se deslocar no turno dele.	-
Perícias	Aponta quais atributos um monstro possui proficiência. Por exemplo, um monstro que seja muito forte possui bônus de perícia no atributo força.	-
Vulnerabilidades, Resistências e Imunidades	As criaturas podem possuir vulnerabilidade, resistência ou imunidade a certos tipos de dano. Por exemplo, um monstro for imune a dano de fogo, nenhum dano de fogo irá afeta-lo.	-

Nível de dificuldade	Representa o nível de desafio de um monstro, ou seja, quão grande é a ameaça que esse monstro representa. Monstros com baixo nível de desafio são insignificantes, exceto em grandes números. Enquanto monstro nível de desafio 21 ou maior representam um grande desafio para os jogadores.	Nível dado pela matriz probabilidade x impacto.
Pontos de experiência	Determina quantos pontos de experiência um monstro concede a ser derrotado, o valor é determinado pelo seu nível de desafio.	Determina quantos pontos de experiência um monstro concede ao ser derrotado, ou completamente evitado ao fim da campanha.
Ações	Descreve diversas ações e comportamentos do monstro	Descreve o risco real.

Fonte: Autor

Na Tabela 1 foi elucidado como os atributos de D&D serviram de metáfora a gamificação. A seguir, é apresentada a metodologia desenvolvida na abordagem para o gerenciamento de riscos. Foram selecionadas as seguintes práticas para o gerenciamento de riscos: I. Identificação de Riscos, II. Análise qualitativa, III. Definição do monstro, IV. Seleção das tarefas e vinculação das tarefas aos monstros e V. Monitoramento do gerenciamento de riscos.

Para a abordagem, definiu-se a utilização de uma ferramenta de identificação de riscos bastante conhecida: *checklist*. Trata-se de uma lista genérica de riscos, que foi pré-configurada na ferramenta, de modo a apresentar um modo geral para identificação de riscos aplicáveis aos times mais inexperientes, o *checklist* em questão foi desenvolvido pelo grupo de pesquisa PENSO no LaCA/UFSM. O *checklist* completo disponibilizado encontra-se disponível no Anexo B. Além da lista pré-configurada, ainda podem ser utilizadas outras ferramentas comuns como *brainstorming*, análise FOFA (Forças, Oportunidades, Fraquezas e Ameaças) ou opinião de especialistas, nesse caso os riscos identificados precisam ser cadastrados na ferramenta. A análise qualitativa deve ser realizada fora da ferramenta.

A definição de monstro é um passo analítico, isto é, para cada risco identificado e priorizado nas etapas anteriores será criado um monstro. O monstro terá um nível, que será dado pela matriz Probabilidade *versus* Impacto, técnica bastante conhecida para o

ranqueamento dos riscos, de forma que é atribuída uma nota para a probabilidade e outra para o impacto, variando de zero a cinco, por fim essas notas são multiplicadas gerando-se assim o nível do monstro. O monstro torna-se por fim uma forma lúdica de representar um risco analisado.

A fase IV, seleção de tarefas, se dá analisando cada um dos monstros criados, bem como escrutinando as tarefas do processo de software. Dessa forma, as tarefas que são executadas e que tem relação com um risco específico passam a serem consideradas missões quando vinculadas a um monstro. As missões podem receber recompensas específicas. Por fim o conjunto de monstro e missões compõe um plano de gerenciamento do risco.

O monitoramento de risco se dá durante todo o jogo, uma vez que o sistema gamificado objetiva aumentar o controle. Dessa forma o gerente de projeto pode acompanhar o desenvolvimento das missões, práticas e tarefas com o objetivo de mitigar ou evitar os possíveis danos causados por um risco, e decidir se um risco saiu do jogo, ou afetou, e nesse último caso, reorganizar o projeto ou decretar sua falha.

Dessa forma, entende-se que as práticas selecionadas para compor a abordagem correspondem a um conjunto de boas práticas, de fácil implantação e que dão conta de forma geral de riscos considerando uma política ativa de gestão de riscos. Sendo elas: I. Identificação de Riscos, II. Análise qualitativa, III. Definição do monstro, IV. Seleção das tarefas e vinculação das tarefas aos monstros e V. Monitoramento do gerenciamento de riscos.

6. VALIDAÇÃO

A metodologia aplicada a validação desse trabalho está dividida em três partes. A primeira etapa constitui-se em uma validação de modelos, em que a abordagem é avaliada quanto a sua adequação a modelos de maturidade e padrões mercadológicos, tais como CMMI, ISO e PMBoK, tal validação é importante principalmente no que tange a aplicação da abordagem desenvolvida. A segunda etapa é uma entrevista com especialistas, a partir de um questionário, a fim de levantar principalmente as opiniões a priori da aplicação da metodologia. E, por fim, a terceira etapa é constituída por uma pesquisa-ação, em que a abordagem é testada em um ambiente real.

Optou-se pelo uso da pesquisa-ação uma vez que essa é sustentada pelo pressuposto que os sujeitos que nela se envolvem compõem um grupo com objetivos e metas comuns, interessados em um problema que emerge num dado contexto em que atuam (PIMENTA, 2005). É importante ressaltar também que o a segunda e a terceira etapas de validação foram desenvolvidas em conjunto com Souza (2019 – no prelo), uma vez que o estudo de Souza (2019 – no prelo) descreve o gerenciamento de projetos gamificado.

6.1 VALIDAÇÃO DE MODELOS TEÓRICOS

Nessa seção, é apresentado como um projeto utilizaria a ferramenta Ranger para realizar a gestão de riscos de forma adequada utilizando-se os processos previstos no PMBoK para a gestão de riscos e SCRUM como gerenciamento do projeto em si. Considerando que o contexto do projeto seja uma equipe pequena composta por dois desenvolvedores, um analista de testes, um *Product Owner* e o *Scrum Master*. A equipe é distribuída geograficamente, isto é, os desenvolvedores trabalham em modo remoto. O projeto apresenta uma taxa de mudança média, sendo executado incrementalmente. Por conta disso, os requisitos definidos podem ser alterados, de forma a adicionar novas funcionalidades.

Além disso o PO conduz semanalmente reuniões com os demais *stakeholders* para a validação dos requisitos. A criticidade do projeto foi definida como perda de dinheiro. Um requisito para ser considerado apto (*definition of ready*) precisa passar pela validação, bem como precisa ter sido priorizado pelo PI e pelos *stakeholders*. E para ser considerado concluído o requisito precisa ter sido desenvolvido, revisado por outro desenvolvedor, e ainda ter sido aprovado nos testes apropriados.

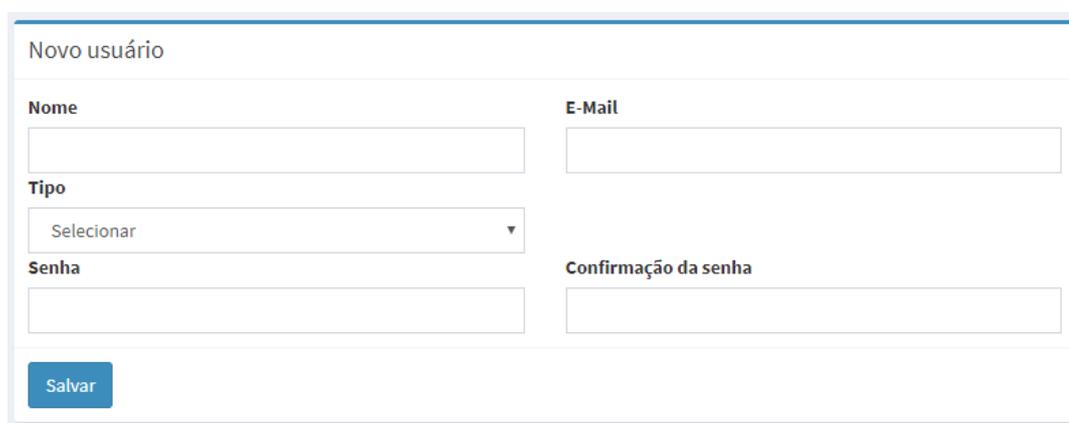
Para esse projeto, o time acordou que a *Sprint* durará duas semanas. Além desse, outros acordos do time são: o *backlog* com suas devidas prioridades será acessível a todos da equipe; e as reuniões de planejamento contarão com a participação de todos os membros do time.

Para que esse projeto seja configurado na ferramenta *Ranger*, será necessário que o *Scrum Master* ocupe o papel de *dungeon master*, uma vez que esse será o responsável pela criação, configuração e manutenção da campanha. Para que seja possível ao *Product Owner* acompanhar, e fazer a manutenção do *backlog* e dos requisitos, a pessoa também terá a permissão de *Dungeon Master*, entretanto, atuará como jogador para as demais atividades, possuindo, dessa forma, uma ficha de jogador. Os demais integrantes do time serão jogadores.

Para inicializar o projeto, o DM deverá acessar na ferramenta o menu *campanhas*, em que selecionará a opção de nova campanha. A campanha aglutinará todas as informações referentes ao projeto, e por conta disso, o DM deverá inserir os dados básicos como o nome, data de início, previsão de prazo e descrição. Após esse processo, o DM adicionará os detalhes da campanha que forem utilizados.

Para a adição de personagens o DM adicionará personagens por meio da opção homônima. Devendo criar contas para os desenvolvedores e analista de teste com o perfil jogador, como pode ser visualizado na Figura 11 - Novo usuário jogador do sistema. O perfil de *Product Owner* necessita da permissão adicional de *Dungeon Master*.

Figura 11 - Novo usuário jogador do sistema



O formulário, intitulado "Novo usuário", contém os seguintes campos:

- Nome:** Campo de texto.
- E-Mail:** Campo de texto.
- Tipo:** Menu suspenso com a opção "Selecionar".
- Senha:** Campo de texto.
- Confirmação da senha:** Campo de texto.

Um botão azul "Salvar" está localizado na base do formulário.

Fonte: Autor

A campanha representa o projeto como um todo, cada uma das *sprints* do projeto é representada por uma aventura. Após a criação e configuração da campanha e dos usuários dos jogadores, o DM procede a criação da primeira aventura pelo menu *Aventuras*. Ao criar

uma nova aventura, além dos atributos padrão, é possível associar monstros, missões e tarefas pré-existentes ou o DM pode criar novas missões, tarefas e monstros.

Utilizando-se do menu *Regras* o *PO* deve estabelecer as regras do projeto em conjunto com os *stakeholders*. Bem como pode verificar se as regras estão associadas corretamente a missões e tarefas.

A partir do *backlog* priorizado, o DM criará no menu *Tarefas* novas tarefas. As tarefas tem como atributos básicos nome, descrição, prioridade e esforço. Além desses atributos, o DM pode associar uma tarefa a um personagem, associar regras, e associar a uma aventura. Ou ainda, se for pertinente, adicionar prêmios para a conclusão da tarefa. Caso as tarefas não pertençam a aventura corrente, o DM pode criá-las da mesma forma, porém sem associar elas a uma aventura.

As contas criadas pelo DM para os jogadores são o meio de acesso desses ao sistema gamificado. No seu primeiro acesso precisam completar a ficha de personagem, que pode ser vista na Figura 12 - Ficha de Personagem.

Figura 12 - Ficha de Personagem

Ficha de personagem

Nome do personagem Classe Nível Experiência

LÓGICA 0

AGILIDADE 0

CREATIVIDADE 0

AÇÃO 0

COMUNICAÇÃO 0

RESILIÊNCIA 0

Habilidades Adicionar

Descrição

Maikovisky

Campanha Histórico

Fonte : Autor

No primeiro acesso, os jogadores configuram sua ficha, iniciando-se no nível 1, com seis pontos para serem distribuídos pelos atributos. Além disso devem selecionar sua classe, e preencher demais informações conforme as regras do sistema.

Para jogar, o DM, associará as tarefas a aventura atual durante a reunião de planejamento, de acordo com o *backlog* da *Sprint*. Bem como associa os personagens dos jogadores as tarefas. Conforme o desenvolvimento das tarefas os jogadores alteram seus status e fazem lançamentos pertinentes no diário da campanha.

Ao completar uma tarefa, o jogador altera o status da tarefa para completado. Diariamente o DM deve verificar os status das tarefas, ao notar uma tarefa completa, deve fazer a avaliação, e, em conjunto com o PO, determinar se a atividade foi completada com sucesso, de acordo com as regras, e dentro do prazo esperado. Caso a tarefa seja concluída com sucesso, o DM atribuirá as recompensas da tarefa ao jogador e a equipe.

Para o exemplo foram identificados riscos relacionados a requisitos a partir da análise do *checklist*, configurando-se como o primeiro processo previsto pelo PMBoK – identificar riscos. Dessa forma, o risco é registrado, e analisado qualitativamente e priorizado – que equivale a outro processo previsto pelo PMBoK – realizar a análise qualitativa de riscos. Nessa etapa ainda é realizada a análise através da matriz probabilidade *versus* impacto, que gerará o nível do monstro. O processo planejar as respostas aos riscos é dado pela fase de identificação e seleção de tarefas relacionadas ao risco. No caso, são selecionadas tarefas do processo e são vinculadas as tarefas ao risco: especificação de requisitos e reunião de validação de requisitos. Ao serem vinculadas ao risco, as tarefas passam a serem tratadas como missões. Sua complexidade será vinculada a complexidade definida qualitativamente para o risco. Com isso definido, o risco é associado a um monstro, e, o conjunto composto pelo monstro e as missões são o plano de risco. As missões serão atribuídas a jogadores nesse caso, porém é possível que sejam atribuídas a grupos de jogadores. De forma a iniciar o processo de Implementar respostas aos riscos, durante a execução, as missões se comportam de forma análoga às tarefas, entretanto, caso ultrapassem os prazos, ou não obtenham resultados, os efeitos dos monstros são ajustados pelo DM. Tratando-se, por fim, do último processo previsto pelo PMBoK: monitorar riscos.

Ao término das missões, os jogadores submetem seus resultados, que são avaliados pelo DM. Em caso de aceite, a missão dá aos jogadores recompensas, e evita o encontro com o monstro. O DM pode, ao final das missões designadas, associar novas missões, caso seja necessário. Entretanto, entendendo que esse risco foi tratado para essa *sprint*, o DM finaliza o monstro retirando ele do jogo. Em caso de um monstro não ser tratado adequadamente e o risco representado ocorrer, a ação do monstro é a obrigação da reorganização das tarefas selecionadas pelo DM. Isto é, um monstro não evitado, e encontrado, como por exemplo, se a especificação de requisitos e validação de requisitos não forem feitas, ou não lograrem seu objetivo, serão reintroduzidas tarefas de codificação que estavam prontas, e serão contadas como retrabalho.

Um exemplo de um plano de risco gamificado na abordagem é para o risco requisitos de baixa qualidade é definido como um monstro, por consequência suas características

precisam ser analisadas, dessa forma se define qual é o seu grau de probabilidade e de severidade, que definirá o nível do monstro. Passo esse feito, devem ser analisado pelo DM quais tarefas do processo comporão o plano de gerenciamento de risco, nesse exemplo, o DM pode decidir por inserir no plano as tarefas confecção de protótipos e reunião de validação de requisitos prototipados. Dessa forma, o plano será composto pelo risco e sua avaliação (monstro) e das tarefas organizadas e vinculadas ao risco (missões).

Considerando que o projeto seja concluído com sucesso, ao final o DM deverá consolidar o histórico das campanhas. Com isso, serão gerados pelo sistema relatórios de tarefas e atividades desenvolvidas, bem como o prazo inicialmente acordado e o prazo final. A partir dos relatórios do sistema, a equipe e a gerência poderão melhorar práticas a partir da análise de riscos ocorridos, medidas que foram consideradas para os planos de risco, e em que grau o processo utilizado necessita ser adequado. Dessa forma, fica evidente que a abordagem prevê a utilização de processos compatíveis com o gerenciamento de riscos no PMBoK.

6.2 QUESTIONÁRIOS A ESPECIALISTAS

Ao analisarmos um estudo em engenharia de software, cada vez mais, temos nos questionado acerca de sua validade. É fundamental que o estudo seja válido para o conjunto de participantes para o qual foi testado, bem como é interessante poder generalizar tais resultados para uma população mais ampla. (Travassos et al, 2002)

Objetivando ampliar a gama de validação, desenhou-se um questionário, foi selecionada uma amostra de especialistas por conveniência, de forma que os selecionados tinham conhecimentos de jogos RPG e em gerenciamento de projetos, podendo, por consequência avaliar de forma mais pragmática um tema. Para tanto foram selecionados dez participantes, com pelo menos cinco anos de experiência em desenvolvimento de software. Os entrevistados responderam questões em três etapas, a primeira era sobre sua experiência com a gestão ou a execução de atividades de projeto tidas como burocráticas, na segunda etapa, as perguntas levantaram temas sobre os perfis envolvidos nos processos de produção de software, e suas características geracionais. Por fim, na última etapa da entrevista, os entrevistados avaliaram a proposta da abordagem *Ranger*. As questões apresentadas compõem o Anexo C dessa dissertação.

Na primeira etapa, que levantava dados da experiência desses entrevistados, dessa forma foi questionado sobre a natureza da organização da gestão dos projetos, percebeu-se que atualmente a maioria dos processos utilizados seguem dinâmicas ágeis, apesar de todos os entrevistados afirmarem que já tiveram experiência em contextos mais planejados ou sem metodologia especificada. Demonstrando que apesar de haver uma tendência de mercado para a aplicação de processos mais ágeis, ainda há casos em que o uso de processos planejados, em escopos específicos.

Dessa forma, foi questionado aos especialistas sobre quais desafios são mais comumente enfrentados por eles e suas equipes em cotidiano laboral. Foram listados desafios localizados na literatura, e foi permitida a seleção de mais de um, entendendo a realidade complexa a qual está submetido o desenvolvimento de software. Dessa forma foram apontados os desafios a seguir como os mais comuns:

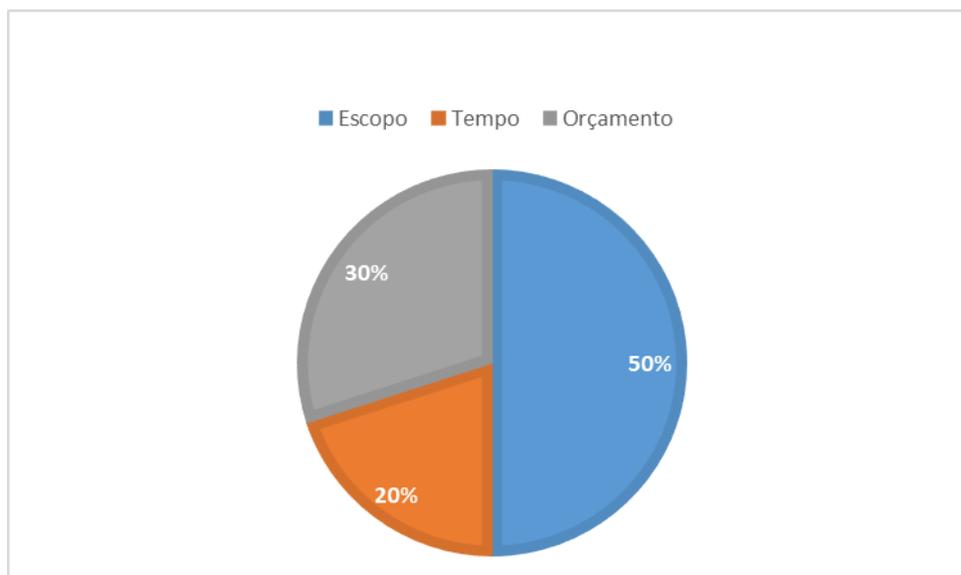
Figura 13 – Maiores desafios de projeto

Fonte: Autor

A Figura 13 demonstra os desafios mais comuns encontrados pelos gestores de projetos. Em mais de 80% das experiências relatadas não havia uma clara gestão de riscos, apesar de em alguns casos haver a identificação de riscos, suas respostas eram, em geral, reativas. Corroborando com o apresentado pela literatura específica. Alguns comentários dos respondentes levam a compreensão de que a gestão de riscos é discutida durante a formação profissional, mas acaba perdendo espaço, por tratar-se de uma atividade bastante burocrática e cujas ferramentas para a execução dos seus processos são bastante complexas e despedem muito esforço intelectual e demandam muito tempo para serem desenvolvidas a contento, e dessa forma acabam por serem deixadas de lado.

Após isso questionou sobre qual aspecto de projeto acontecem frequentemente problemas, isso é, questionou-se se os problemas mais recorrentes eram vinculados ao escopo, ao custo ou ao prazo em primeira análise. Os maiores problemas relatados pelos entrevistados são relativos ao escopo do projeto, conforme é visto na Figura 14, que demonstra a área de vinculação do projeto, o que implica no prazo e no custo. Foram apontados como falhas de gerenciamento de projeto mais preponderantes a análise de requisitos, inexistência de critérios de aceitação, estimativas equivocadas e falta de gerenciamento de riscos.

Figura 14 – Áreas de falha de projetos



Fonte: Autor

Na segunda etapa, os respondentes foram questionados sobre características de suas equipes. A faixa etária das pessoas que compõe as equipes de que participam nossos entrevistados variam de 22 a 49 anos, o que coloca três gerações distintas lado a lado no mercado de trabalho. Os entrevistados foram questionados se são perceptíveis diferenças entre as gerações no seu acesso, e execução de atividades laborais relativas ao planejamento, documentação, avaliação, testes, e demais atividades que não sejam diretamente a codificação, e todos os entrevistados apontaram que há diferenças perceptíveis. A seguir, questionou-se, utilizando-se da experiência empírica dos entrevistados, se esses acreditavam que para suas equipes seria interessante a utilização de uma abordagem gamificada para o gerenciamento de seus projetos. Os respondentes indicaram que há diferenças entre as faixas etárias e seus interesses, entretanto que, em sua maioria, os membros das equipes demonstram interesses em soluções gamificadas.

Por fim, a última seção de questões abertas apresentava a abordagem *Ranger* completa, isto é, gerencia de projetos e de riscos gamificadas, e solicitava aos respondentes uma avaliação a luz dos processos que esses executam diariamente, bem como sua experiência. As avaliações foram feitas em uma pergunta aberta, os participantes demonstraram interesse na abordagem, ressaltaram que diversas vezes as atividades de engenharia de software são deixadas de lado justamente por serem consideradas burocráticas e pouco profícuas para o desenvolvimento da solução, e a abordagem *Ranger* poderia engajar suas equipes a desenvolverem tais atividades.

Também foi apontado que como a maioria dos projetos apresenta uma técnica reativa de controle de riscos, a aplicação da abordagem colaboraria para uma mitigação ativa dos riscos, uma vez que a abordagem Ranger trata explicitamente dessa gestão ativa. Quando questionados sobre a aceitabilidade da ferramenta, os respondentes denotaram certo interesse, contrapondo que a adição de práticas poderia ser visto como apenas um aumento na carga laboral da equipe. Assim, entende-se que para a implantação da abordagem, é necessário um processo de sensibilização e formação da equipe quanto a real necessidade da aplicação de práticas preditivas de riscos.

6.3 ESTUDO DE CASO

Por fim, a validação da abordagem Ranger foi pensada inicialmente como pesquisa-ação. Essa metodologia foi selecionada em detrimento do estudo de caso por entender que a principal diferença entre ambas é que na pesquisa-ação, o pesquisador propõe uma melhoria, acompanha sua implantação e faz sua avaliação como parte do processo científico, enquanto na metodologia de estudo de caso o pesquisador observa um evento, o descreve e avalia, sem alterá-lo. (Tripp, 2005)

Para Tripp (2005), a pesquisa-ação é composta por quatro fases cíclicas, a primeira sendo o planejamento da melhoria a ser implementada; a segunda a ação para a implantação da melhoria; a terceira o monitoramento e a descrição dos efeitos da ação; e a última a avaliação dos resultados da ação.

Seguindo a metodologia proposta por Tripp (2005), a primeira fase sendo o planejamento englobou a elaboração da abordagem, bem como, o desenvolvimento da ferramenta *Ranger* que seria utilizada para a validação da abordagem. Entretanto, ao aplicar-se a metodologia, acabou-se por impossibilidade da participação efetiva do pesquisador por utilizar-se de um estudo de caso, em que o pesquisador guiou a aplicação de uma melhoria de processo e analisou sua implantação.

Após o planejamento da melhoria, passou-se a ação de implantação. Para tanto, os desenvolvedores foram treinados pelos pesquisadores para que compreendessem a abordagem proposta, bem como conseguissem fazer a utilização da ferramenta Ranger em uma reunião prévia. Nessa reunião, ficou também acertado que qualquer dificuldade encontrada seria relatada, bem como o pesquisador fariam alguns acompanhamentos contínuos quanto ao uso da ferramenta pelos sujeitos da pesquisa.

Uma vez que o projeto de software em questão apresenta um prazo ainda longo para a sua conclusão, o pesquisador tem feito o monitoramento da aplicação da melhoria através de questionários e entrevistas não estruturadas, ao passo que não foi possível concluir o processo com a avaliação completa, uma vez que o projeto não foi encerrado.

O projeto em acompanhamento iniciou-se, e os procedimentos da abordagem têm sido observados. Na inicialização do projeto a equipe apresentou diversas dúvidas com a gerência gamificada, principalmente, pois não havia uma figura clara na equipe de liderança. Ao definir essa responsabilidade, alguns gargalos foram gerados, uma vez que o Dungeon Master também atuava como analista e desenvolvedor, principalmente tendo em vista a pequena equipe. Por consequência, houve por parte desse uma pequena resistência a

implantação da melhoria. Após esse processo inicial, uma vez que a abordagem explicita a necessidade do gerenciamento ativo de riscos, que não era prática costumeira, tais atividades geraram algum desconforto, entretanto, pelo aspecto gamificado, sua adesão foi facilitada.

Durante a primeira aventura definida, foram identificados os riscos, selecionadas tarefas e missões e foi dado o início. Por se tratar de uma metodologia ágil, poucas tarefas foram adicionadas à aventura, e a aventura terminou dentro do previsto. Uma vez que é um projeto com dedicação de poucas horas semanais, implicou em uma aventura mais longa.

Os resultados parciais têm demonstrado que as tarefas de engenharia têm sido desenvolvidas corretamente, dentro dos prazos esperados, rendendo assim a mitigação dos riscos, e por consequência o atendimento dos requisitos, do prazo e do custo. Os relatos dos sujeitos da pesquisa corroboram para o aumento da motivação, uma vez que os sujeitos afirmam terem sentido interessados em terminarem o conteúdo gamificado. Em contrapartida, os sujeitos também afirmaram que a utilização da abordagem insere outras atividades que devem ser respondidas.

Nota-se com isso que ainda está pendente a finalização da pesquisa, com uma avaliação mais quantitativa sobre a produção, que depende da finalização do projeto para que possa ser completamente aplicada. Nas entrevistas não-estruturadas que ocorreram durante o desenvolvimento, os desenvolvedores apontaram que se sentem mais motivados, por entenderem o processo mais completamente, bem como por verem uma progressão a partir das atividades de engenharia.

Em virtude disso, a luz dessa avaliação parcial, entende-se que a abordagem proposta atendeu aos objetivos iniciais de aumento da motivação, bem como a melhoria dos processos e a aplicação de modelos de gerenciamento ativo de riscos.

7. CONSIDERAÇÕES FINAIS

A produção de software atualmente enfrenta diversos desafios como a manutenção de custos, o cumprimento de prazos, e o atendimento das necessidades do cliente são alguns desses que historicamente acompanham o desenvolvimento, entretanto a chegada de novas gerações, que tem visões diferentes de mundo, trouxeram para a produção de software novos desafios. Em síntese, com a caminhada para processos mais ágeis, e com a complexidade inerente do gerenciamento de riscos, bem como a chegada das novas gerações ao mercado de trabalho, o gerenciamento de riscos passou a ser considerado uma atividade burocrática e em muitos contextos deixada como uma segunda preocupação.

Esse trabalho apresentou uma abordagem gamificada para o gerenciamento de riscos, incluído dentro de um processo gamificado de gerenciamento de projetos. A abordagem é composta por monstros, missões e recompensas, objetivando um gerenciamento ativo dos riscos, podendo ser esses evitados ou mitigados através da utilização da abordagem.

A abordagem proposta foi validada através de sua implementação na ferramenta Ranger. A validação contou de três etapas, uma teórica, que avaliou a compatibilidade da abordagem com padrões de mercado, a segunda que consistia de uma entrevista, em que especialistas colaboraram com a pesquisa com seu conhecimento empírico da área de produção de software, e por fim, a terceira sendo uma pesquisa ação, em que houve o planejamento de uma melhoria, a ação de implantação da melhoria e o acompanhamento e a avaliação da melhoria.

Consideramos, dessa forma, que foi desenvolvido um metamodelo que estruturou os elementos da abordagem gamificada para o gerenciamento de riscos em projetos de software, bem como foram identificados os relacionamentos entre práticas de processos de referência como o PMBoK e o processo gamificado. Foi selecionado um *checklist* de riscos para o processo de identificação de riscos. Bem como, por fim, foram selecionadas no escopo da ferramenta um conjunto de boas práticas, que aplicadas corretamente podem colaborar para a gestão ativa de riscos.

Por fim, apesar de haver ainda pontos a serem estudados, e diversas melhorias a serem implementadas em estudos futuros, acredita-se que a proposta desse trabalho cumpriu com seus objetivos iniciais. Principalmente diante das dificuldades da aplicação de métodos experimentais para a validação de abordagens de gerenciamento de projetos, e a impossibilidade de generalização de regras para pessoas.

Essa dissertação apresenta como principal contribuição a abordagem gamificada de gerenciamento de riscos, utilizando-se da metáfora de Dungeons & Dragons. Bem como o protocolo de validação via pesquisa-ação, e a extensão da ferramenta *Ranger* para atender a abordagem.

Para trabalhos futuros, pretende-se completar o processo de validação no contexto descrito aqui, bem como ampliar a validação para outros contextos de projetos.

REFERÊNCIAS

ABNT NBR ISO 31000:2009

ABNT IEC 31010, Risk management – Risk assessment guidelines

ABNT ISO GUIA 73:2009, Gestão de riscos – Vocabulário.

BELINAZO, G. e FONTOURA, L. M. Mapeamento sistemático de Gamificação em Engenharia de Software. 1ª Escola Regional de Engenharia de Software - ERES. p 191-200, Alegrete-RS, 2017

BOEHM, B.W.: Software risk management: principles and practices. IEEE Software 8 (1), 32–41. (1991).

BZUNECK, J.A. (2004). A motivação do aluno: aspectos introdutórios. Em: E. Boruchovitch e J.A.

CATANIA, A.C. (1979). Learning. Englewood Cliffs: Prentice-Hall.

DUBOIS, D. J.; TAMBURRELLI, G. Understanding gamification mechanisms for software development. In: acm. Proceedings of the 2013 9th joint meeting on foundations of software engineering. [s.l.], 2013. P. 659–662.

KUMAR, J. (2013) Gamification at work: designing engaging business software. In: Marcus A. (eds) design, user experience, and usability. Health, learning, playing, cultural, and cross-cultural user experience. Duxu 2013. Lecture notes in computer science, vol 8013. Springer, Berlin, Heidelberg

KATZELL, R. A.; Thompson, D. E. Work motivation: theory and practice. American Psychologist, v. 45, n. 2, p. 144-153, 1990.

LATHAM, M., Constructing the team. HMSO London. 1994

MEDEIROS, D. B., Neto, P. A. S., Passos, E. B., Araújo, W. S. . Working and Playing With Scrum. In International Journal of Software Engineering and Knowledge Engineering. 2015.

SCHÖNBOHM, A. JÜLICH, A. On the Effectiveness of Gamified Risk Management Workshops: Evidence from German SMEs (2016)

MENEZES JUNIOR, J., GUSMÃO, C. E MOURA, H. Risk factors in software development projects: a systematic literature review. in Software Quality Journal. 2018.

MEARLS, M. Dungeon Master's Guide. Wizards of the Coast LLC. 2014. Renton, WA.

MEARLS, M. Player's Handbook. Wizards of the Coast LLC. 2014. Renton, WA.

NIETZSCHE, F. La volonté de puissance. Paris: NRF, 1970.

PEDREIRA, O; GARCÍA, F; BRISABOIA, N Mario PIATTINI. (2015) Gamification in software engineering – A systematic mapping. In Information and software technology

- PMI, "PMBOK Guide: A guide to the Project Management Body of Knowledge," 2017.
- PRESSAMAN, R. S., & MAXIM. B.R. Engenharia de Software: Uma abordagem profissional. 8ª Ed. Porto Alegre. AMG. 2016.
- STEPHEN N. Luko (2013): Risk Management Terminology, Quality Engineering, 25:3, 292-297
- WALLACE, L., KEIL, M. e Rai, A. Understanding software project risk: a cluster analysis. Information & Management. Volume 42, Issue 1 p 115-125. 2004.
- WAGNER, S. e RUHE, M. A Systematic review of productivity factors in software development. 2nd International Workshop on Software Productivity Analysis and Cost Estimation. 2018.
- WILLIAMS, L.; COCKBURN, A. Agile Software Development: It's about Feedback and change. Computer, New York. v. 36, n. 6, p.39-43, june, 2003.
- SOUZA, D. A. Z. de. Uma abordagem gamificada para o gerenciamento de projetos de software. Dissertação de mestrado. 2019, no prelo.
- DEEMER, P. BENEFIELD, G. LARMAN, C. VODDE, B. Scrum Primer: A Lightweight Guide to the Theory and Practice of Scrum. 2017.
- PIMENTA, S. G.; Pesquisa-ação crítico-colaborativa: construindo seu significado a partir de experiências com a formação docente. Educação e Pesquisa, São Paulo, v. 31, n. 3, p. 521-539, set./dez. 2005
- SARIGIANNIDIS, L. e CHATZOGLU, P. D. Quality vs Risk: An investigation of their relationship in software development projects. IN: International Journal of Project Management 32 (2014) p.1073-1082. 2013.

ANEXO A – Histórias de usuário do sistema

US001 - Elaborar ficha de personagem

Eu como um jogador

Quero elaborar a minha ficha de personagem

Para que eu consiga participar da campanha

US002 - Definir classe de personagem

Eu como jogador

Quero definir a classe do meu personagem

Para determinar meu papel no projeto

US003 - Definir habilidades do personagem

Eu como jogador

Quero definir quais as habilidades que eu tenho proficiência

Para que eu consiga utilizá-las durante a campanha

US004 - Distribuir pontos de atributo do personagem

Eu como jogador

Quero distribuir os pontos de atributos no meu personagem

Para que eu possa caracterizar adequadamente minhas aptidões físicas e mentais

US005 - Participar de um grupo de campanha

Eu como jogador

Quero fazer parte de um grupo

Para que eu possa identificar a minha equipe na campanha

US006 - Escolher tarefas

Eu como jogador

Quero escolher tarefas que eu atenda aos requisitos

Para executá-las, evoluir e ganhar recompensas

US007 - Consultar mapas

Eu como jogador

Quero consultar os mapas

Para que eu consiga me guiar na campanha

US008 - Escrever diário da campanha

Eu como representante dos jogadores

Quero escrever o diário da campanha

Para documentar a visão dos personagens

US009 - Consultar fichas de personagem

Eu como um jogador ou dungeon master

Quero consultar uma ficha de personagem

Para visualizar suas características e feitos

US0010 - Consultar histórico da aventura

Eu como um jogador ou dungeon master

Quero consultar o histórico de uma aventura

Para verificar o andamento da mesma

US011 - Consultar histórico da campanha

Eu como um jogador ou dungeon master

Quero consultar o histórico de uma campanha

Para verificar o andamento da mesma

US012 - Consultar regras e restrições

Eu como um jogador ou dungeon master

Quero consultar regras e restrições de tarefas

Para verificar as necessidades da mesma

US013 - Definir o mundo

Eu como um dungeon master

Quero definir o mundo de uma ou várias campanhas

Para conseguir contextualizar os elementos que irão fazer parte da experiência de jogo

US014 - Definir a campanha

Eu como um dungeon master

Quero definir uma campanha

Para conseguir executar posteriormente

US015 - Definir aventura

Eu como um dungeon master

Quero definir uma aventura

Para conseguir executar a mesma posteriormente

US016 - Definir tarefa

Eu como um dungeon master

Quero definir uma tarefa

Para que eu consiga disponibilizar a mesma para os jogadores

US017 - Definir regras e restrições

Eu como um dungeon master

Quero definir as regras e restrições de uma campanha

Para que eu a campanha esteja de acordo com o projeto

US018 - Vincular uma regra a uma tarefa

Eu como um dungeon master

Quero vincular uma regra a uma tarefa

Para que a mesma seja executada de acordo com os requisitos estabelecidos

US019 - Estabelecer requisito de habilidade

Eu como um dungeon master

Quero estabelecer um requisito de habilidade para uma tarefa

Para que a tarefa seja executada por um personagem apto

US020 - Estabelecer recompensa de tarefa

Eu como um dungeon master

Quero estabelecer uma recompensa para uma tarefa

Para que eu possa recompensar os executores da mesma

US021 - Estabelecer recompensa de marco

Eu como um dungeon master

Quero estabelecer uma recompensa por um marco de campanha ou individual

Para que eu possa recompensar os jogadores de acordo com seus feitos

US022 - Estabelecer um grupo

Eu como um dungeon master

Quero escolher os personagens que irão fazer parte de um grupo

Para determinar os grupos que irão executar uma campanha

US023 - Editar uma campanha

Eu como um dungeon master

Quero editar uma campanha

Para adequar as necessidades

US024 - Selecionar monstros

Eu como dungeon master

quero selecionar monstros

para representar riscos

US025 - Criar monstros

Eu como dungeon master

quero criar monstros

para representar riscos

US026 - Selecionar missões

Eu como dungeon master

quero escolher missões para contornar cada um dos monstros selecionados

US027 - Criar missões

Eu como dungeon mestre

quero criar missões para contornar cada um dos monstros selecionados

US028 - Acompanhar a campanha

Eu como jogador/dungeon master

quero acompanhar e saber quais monstros estão em tabuleiro e quais missões estão sendo feitas para evitar os seus encontros

US029 - Gerenciar a campanha

Eu como dungeon master

quero gerenciar o andamento da campanha

US030 - Definir poder de monstro

Eu como dungeon master

quero definir o poder de um monstro para o contexto do projeto

US031 - Sugerir monstro

Eu como jogador

quero sugerir monstros conforme vejo o desenvolvimento do projeto

US032 - Segeir monstro

Eu como dungeon master

Quero receber sugestões de monstros que podem entrar em jogo a partir de threshold da execução do projeto, ou dos jogadores e de práticas de sucesso

US033 - Histórico de campanha

Eu como dungeon master

quero analisar o histórico da campanha e marcar práticas de sucesso

US034 - Definição de threshold

Eu como dungeon master

quero definir limites aceitáveis para variáveis relativas ao projeto

US035 - Finalização da campanha

Eu como dungeon master

quero finalizar a campanha, tirar os monstros do tabuleiro, e definir o sucesso.

US036 - Atribuir missões

Eu como dungeon master

quero atribuir missões a jogadores ou a grupos de jogadores.

ANEXO B – Riscos pré-carregados no sistema

RISCOS

Aprovação inadequada de documentos por parte do cliente
Cliente insatisfeito
Cliente que defini requisitos, aprova-os e sai repentinamente do projeto
Disponibilidade de fundos do cliente
Falhas ao gerenciar as expectativas do usuário final
Habilidade e experiência do cliente
Produto final não corresponde as expectativas do cliente
Qualquer problema com o cliente, tais como: demora na aprovação de documentos, comunicação pobre, resistência à mudança, falta de comprometimento, falta de cooperação, conflitos entre clientes e departamentos.
Resistência a mudanças por parte dos empregados do cliente
Responsabilidades por danos diretos aos clientes
Treinamento inadequado dos usuários
A Equipe de desenvolvimento não tem incentivo para escrever o software de maneira que seja fácil de manter.
Adicionar mais funcionalidades no software a ser mantido que o especificado / dar extras ao cliente (gold-plating)
Algoritmos especificados não estão satisfazendo os requisitos de funcionalidade
Alta rotatividade dentro da organização usuária.
Ambiente de desenvolvimento diferente do ambiente de implantação
Ambiente improdutivo
Ambientes de integração e testes inadequados
Atitude do “é possível” dos gerentes sem antes consultar o time
Avanço do projeto sem aprovação do cliente
Baixa moral / falta de compromisso devido ao baixo nível de entusiasmo afetando assim a produtividade e criatividade. As pessoas não sentem reconhecidas e recompensadas pelos superiores.
Começo da fase de design/desenvolvimento com requisitos em aberto
Controle do produto inadequado.
Decisões técnicas do projeto afetadas por decisões políticas
Deficiência nos componentes fornecidos externamente
Desenvolvimento de interface do usuário inadequada
Dificuldade de uso por causa de um design da interface pobre
Dificuldade em gerenciar mudanças de prioridades. Priorizar soluções emergenciais para solucionar os problemas em detrimento de alternativas mais robustas.
Dificuldade em obter cenários e dados de testes realísticos
Dificuldades em conseguir conselho de um expert no sistema de desenvolvimento

Equipe com carência de pessoas

Equipe inexperiente

Equipe não trabalha junto sem a interferência da gerência

Falha na comunicação entre o time

Falha na coordenação de ambientes de desenvolvimento distribuídos

Falta de aderência aos padrões de desenvolvimento de software

Falta de comprometimento da gerência sênior (alta gerência).

Falta de controle ou plano formal para todas as atividades de desenvolvimento

Falta de documentação e suporte para o processo de desenvolvimento

Falta de experiência prévia com o sistema de desenvolvimento

Falta de orientação sobre a qualidade do produto

Falta de padrão de programação durante o ciclo de vida do software.

Falta de planos de contingência para os riscos conhecidos

Falta de uma notificação formal de eventos que afetam o trabalho

Falta espírito de equipe.

Fraco monitoramento do progresso das atividades

Gerência de Configuração inadequada.

Gerentes inexperientes

Gerentes não consultam o time antes de tomar decisões que afetam seus trabalhos

Inadequada preparação e realização dos testes.

Informações incorretas para relatórios de progresso

Linguagem de programação pode se tornar obsoleta

Linguagem inadequada para o desenvolvimento do software

Manutenibilidade não é incorporada no processo de desenvolvimento de software.

Membros do time desconfortáveis com o modelo de desenvolvimento

Métricas inadequadas / inexatas.

Modelo de desenvolvimento inadequado

Mudanças no time

Muitos designs novos no projeto

Não conseguir contratar recursos humanos especializados

Nenhum treino é feito para o uso das ferramentas de desenvolvimento

Os dados processados pelo sistema são armazenados em diferentes arquivos, que podem ter estruturas incompatíveis.

Os modelos, processos, padrões, métodos e ferramentas de apoio selecionados são inexistentes / inadequados para o processo de manutenção de software.

Partes da implementação do produto não estão completamente definidos pelas especificações do design

Perda de apoio da alta administração devido à mudança de enfoque ou mudança de pessoas

Pessoas envolvidas nas fases iniciais do projeto saírem da empresa

Problemas com software usados durante o desenvolvimento do projeto

Problemas na integração/interface entre sistemas

Projeto não é gerenciado conforme o plano

Recursos de software ou hardware insuficientes para a equipe do projeto

Recursos e desempenho do sistema inadequados

Releases instáveis

Requisitos de desempenho do software a ser mantido não atendidos.

Requisitos de segurança e confiança não satisfeitos

Retrabalho

Software depende de pessoas chave do time

Software entregue com qualidade baixa

Subcontratação

Tecnologia nova / imatura (uso de novas tecnologias)

A área do programa afetado é crítica.

A execução das mudanças impacta funcionalidades de outros sistemas.

A execução de mudanças impacta funcionalidades do próprio sistema.

Alto custo para testar uma mudança em termos de orçamento e tempo.

Alto nível de complexidade técnica das mudanças solicitadas.

Características de hardware não confiáveis ou um suporte técnico do fabricante, dificultando a identificação da fonte de um problema do software a ser mantido.

Código do programa a ser mantido complexo e desestruturado.

Dificuldade dos mantenedores em entender os truques de programação.

Dificuldade em manter sistemas de tempo real e sistemas altamente sincronizados.

Dificuldade em mensurar desempenho necessário para a manutenção do software.

Documentação / papelada excessiva referente ao software a ser mantido

Erro ao estimar (tempo, custo e esforço) necessário para a manutenção do software.

Estrutura corrompida do software a ser mantido.

Falha ao executar software a ser mantido.

Falta / insuficiência de controle do processo de manutenção de software.

Falta de metodologia efetiva para manutenção de software

Falta de procedimentos, programas adequados e recursos para assegurar a garantia da qualidade ao manter software.

Falta de suporte para reengenharia do software a ser mantido

Grande número de funções principais que são afetadas pela mudança.

Grande número de linhas de código afetadas com a mudança.

Inadequadas especificações de mudanças.

Mudanças contínuas no objetivo e escopo ao manter software.

Mudanças no design durante o desenvolvimento

Mudanças no sistema de hardware e software.

O entendimento do sistema a ser mantido é limitado.

O sistema de manutenção não possui capacidade suficiente.

Planejamento inapropriado, incluindo construção e atualização do plano de contingência do software a ser mantido.

Pobre análise de impacto do software a ser mantido e da organização com relação às habilidades críticas, documentação e processos da área.

Pobre integridade dos dados do software a ser mantido

Pobre suporte ao software que está sendo mantido.

Pouca confiança no sistema de manutenção devido à indisponibilidade dos componentes ou erro / mau funcionamento dos componentes / subprojetos.

Pouca ou nenhuma documentação. O software a ser mantido não possui documentação ou quando a documentação existente é insuficiente.

Produto integrará um sistema existente

Tecnologias e sistemas a serem mantidos, antiquados.

Condições de quebra e manutenção de contrato

Crises econômicas

Desalinhamento com os objetivos do negócio

Diferenças legais entre países do cliente e da companhia

Dificuldade em adaptar rapidamente as mudanças referentes ao ambiente de negócios.

Estratégia organizacional adotada.

Falta de maturidade / instabilidade organizacional

Falta de solução para quebra de contrato

Fatores políticos (companhia, clientes, contratantes associados e sub-contratantes) que causam problemas para a manutenção do software.

Fraca interação (comunicação) do gerente a todos com os envolvidos do projeto.

Garantia insuficiente

Impostos locais e nacionais

Inconsistência nas regulamentações dentro do país ou organização

Inovação pelo concorrente

Leis e códigos locais/estaduais

Mudanças nas leis

Os papéis e as responsabilidades de relacionamentos mal definidos ou mal entendidos.

Oscilação da taxa de câmbio

Problemas relacionados aos contratos associados.

Rompimento da empresa

Ausência de um líder.

Ausência de um processo de análise de requisitos bem definido

Baixa qualidade do projeto.

Cronograma irreal

Dificuldade em medir esforço.

Estimativas irreais ou instáveis

Falta de consciência dos riscos

Interfaces externas entre produto e sistemas externos indefinidas ou mal definidas

Interfaces internas entre software e hardware indefinidas

Orçamento irreal

Sem fundos para treinamento e tutoria

Tempo inadequado para realização de testes

Análise inadequada quando um novo requisito é adicionado

Cliente e equipe têm visões diferentes de um mesmo requisito

Cliente incerto sobre os requisitos

Escopo / objetivos pouco claros ou equivocados

Falta de um acordo interativo entre cliente e os mantenedores relativos às especificações dos requisitos.

Grande Backlog

Limites de hardware limitam os requisitos de usuário

Mudança constante nas funcionalidades do sistema

Problemas de desempenho de tempo real (há tempos de respostas restritos).

Projeto de um tamanho/complexidade nunca feito antes

Requisitos de baixa qualidade

Requisitos de segurança são mais complexos que o estado-da-arte

Requisitos especificam algo nunca feito antes pelo contratante

Requisitos funcionais podem não apresentar design possível

Requisitos incompletos do software a ser mantido.

Requisitos instáveis do software a ser mantido.

Requisitos mal entendidos (não refletem as expectativas do cliente).

Requisitos não claros (ambíguos / imprecisos).

Restrições referentes ao hardware designado

ANEXO C – QUESTIONÁRIO

Pesquisa

Essa pesquisa é parte de um trabalho desenvolvido junto ao Laboratório de Computação Aplicada - LaCA - da Universidade Federal de Santa Maria sobre Gamificação e Engenharia de Software.

O questionário está dividido em três sessões, a primeira deve ser respondida diretamente, para a segunda a partir da leitura da descrição inicial e a terceira em base as informações da ferramenta.

1. O que você entende por gamificação?

2. Quais são os desafios mais comuns do desenvolvimento de software para você?

- Falta de planejamento;
- Falha em requisitos de projeto;
- Ausência de controle de qualidade;
- Determinações mal feitas ou incompletas;
- Inexistência de critérios que determinam a aceitação e a qualidade;
- Falta de padronização e/ou metodologias e/ou processos;
- Estimativas equivocadas;
- Falta de inspeções;
- Falta de gerenciamento de riscos;

3. Quais dos aspectos a seguir apresenta maior número de falhas na sua experiência de projeto?

- Escopo
- Tempo
- Orçamento

Responda as seguintes questões considerando que gamificação é o uso de técnicas de jogos para cativar pessoas por intermédio de desafios constantes e bonificações. Considerando sua relação com o trabalho de desenvolvimento de software, bem como a dos seus colegas e chefes.

4. Considerando o conceito de gamificação apresentado, você já observou sua utilização? Se sim, em quais contextos?

5. Considerando a geração X, isto é, aquela que nasceu entre 1960 a 1979. Você acredita que desafios constantes e bonificações aplicadas ao trabalho possam melhorar o engajamento das pessoas em um processo de desenvolvimento de software?

- Concordo completamente.
- Concordo parcialmente.
- Discordo parcialmente.
- Discordo completamente.

6. Considerando a geração Y, isto é, aquela que nasceu entre 1980 a 1990. Você acredita que desafios constantes e bonificações aplicadas ao trabalho possam melhorar o engajamento das pessoas em um processo de desenvolvimento de software?

- Concordo completamente.
- Concordo parcialmente.
- Discordo parcialmente.

(). Discordo completamente.

7. Considerando a geração Z, isto é, aquela que nasceu entre 1995 a 2010. Você acredita que desafios constantes e bonificações aplicadas ao trabalho possam melhorar o engajamento das pessoas em um processo de desenvolvimento de software?

(). Concordo completamente.

(). Concordo parcialmente.

(). Discordo parcialmente.

(). Discordo completamente.

8. Considerando sua experiência, o conceito de gamificação, e as gerações X, Y e Z. Você acredita que a adição de gamificação às atividades de desenvolvimento de software pode apresentar melhoria dos resultados do processo de desenvolvimento?

(). Concordo completamente.

(). Concordo parcialmente.

(). Discordo parcialmente.

(). Discordo completamente.

Por fim, considerando o conceito de gamificação exposto anteriormente, bem como entendendo que a gamificação pode contribuir para o gerenciamento de projetos, responda as seguintes questões:

9. Você utilizaria um sistema gamificado para o gerenciamento de projetos?

(). Concordo completamente.

(). Concordo parcialmente.

(). Discordo parcialmente.

(). Discordo completamente.

10. Considerando a seguinte breve apresentação de uma ferramenta de gestão de projetos, que utiliza-se de uma metáfora do jogo *Dungeons & Dragons*, em que é possível adaptar diferentes processos, de forma a dar maior controle ao gerente de projeto, bem como dar maior conhecimento dos processos a equipe, bem como melhorar *feedbacks*, e, ainda, trazer elementos de conforto do conhecimento. Como você entende que tal ferramenta seria aceita pela sua equipe?

11. Considerando suas práticas em relação a gestão de riscos em seu projeto corrente, você entende que usa práticas reativas ou proativas?

12. Considerando a ferramenta mencionada na questão 10, e considerando que a ferramenta também está integrada com práticas proativas de gerenciamento de riscos, que são compatíveis com práticas prescritas por processos padrão de mercado, você faria uso de tal ferramenta na íntegra? Como acredita que seria aceita tal ferramenta pelos usuários?