

Em Direção à Extensão do *Kernel Essence* para o Gerenciamento de Projetos de Jogos Digitais

Lucas Pieva¹, Raquel Pillat¹, Maicon Bernardino¹

¹Laboratory of Empirical Studies in Software Engineering (LESSE)
Graduate Program in Software Engineering (PPGES)
Universidade Federal do Pampa (UNIPAMPA)
Av. Tiarajú, 810, Ibirapuitã – Alegrete, RS – Brasil

{lucaspieva.aluno, raquelbasso}@unipampa.edu.br, bernardino@acm.org

Abstract. *Context:* The creative industry is thriving and growing on a large scale. In this regard, it is essential to enhance the game development life cycle based on best practices in Digital Game Project Management (DGPM) to achieve successful projects within the Triple Constraint (costs, time, and scope). *Objective:* This study aims to propose an extension of the Kernel Essence for DGPM. *Method:* We propose our theoretical model based on preliminary empirical evidence from the game industry by using action research. *Result:* The theoretical model of DGPM encompasses areas of concerns, alphas, their states, and competencies. *Conclusion:* Although the model has not yet been evaluated, we are confident that we are moving towards our proposed objective.

Resumo. *Contexto:* A indústria criativa está em alta e crescendo em grande escala. Nesse sentido, é essencial melhorar o ciclo de vida de desenvolvimento de jogos com base nas melhores práticas de Gerenciamento de Projetos em Jogos Digitais (GPJD) para alcançar um projeto de sucesso na Tríplice Restrição (custos, tempo e escopo). *Objetivo:* Propor uma extensão do Kernel Essence para o GPJD. *Método:* Propomos nosso modelo teórico com base em evidências empíricas preliminares com a indústria de jogos, usando pesquisa-ação. *Resultado:* O modelo teórico de GPJD compreende as áreas de interesse, os alfas e seus estados e competências. *Conclusão:* Apesar de ainda não termos avaliado o modelo, estamos confiantes de que estamos em direção ao objetivo proposto.

1. Introdução

É notória a evolução tecnológica apresentada pela computação moderna, seja pela miniaturização de componentes, a constante avaliação da assertividade da Lei de Moore ou a abrangência da utilização da inteligência artificial no cotidiano. Esta evolução ocorre pela utilização organizada de métodos científicos na construção de conhecimento e gestão de soluções. Para o desenvolvimento de softwares, a área da computação que estuda as melhores práticas e soluções para a gestão dos processos e organização sistematizada das atividades é a Engenharia de Software (ES). Assim, tendo o *software* se inserido nas mais diversas áreas de atuação, a ES necessita de constante evolução para obter êxito nos desafios de integrar seus métodos no Gerenciamento de Projetos (GP) [Kerzner 2021].

Dentre as áreas que geram dificuldades para o GP, podemos destacar o desenvolvimento de jogos digitais (JDs), pois a interdisciplinariedade técnica-criativa aliada

à infinidade de dispositivos e plataformas alvo podem tornar o projeto muito complexo, podendo causar a ruína do projeto no seu fundamento principal, que é divertir e imergir o jogador. Poderemos verificar ao longo desta pesquisa que há inúmeros estudos, sendo [Pandey et al. 2018], [Luhova et al. 2019], [Kristiadi et al. 2019], [Oliveira et al. 2011] e [Améndola et al. 2015] exemplos desta vasta gama de iniciativas de aliar a ES ao Gerenciamento de Projetos em Jogos Digitais (GPJD), cada uma utilizando seu próprio método, ficando então compreendido não haver uma melhor prática adotada para este propósito.

Para atender a possibilidade de uniformizar a modelagem dos elementos essenciais de um empreendimento de *software*, surge a iniciativa produzida pela *Software Engineering Methods and Theory* (SEMAT). Esta solução compreende um conjunto de elementos textuais e gráficos, com sintaxe e semântica própria, para modelagem do que é essencial de um *Software*, independente da abordagem de GP. Tal solução foi nomeada como *Essence* [SEMAT 2018].

O *Essence* apresenta um núcleo (*Kernel*) com alguns artefatos para compor o conjunto base de qualquer Empreendimento de Software, além de apresentar uma linguagem própria para escrita de elementos complementares à modelagem sintaticamente correta de processos, métodos, metodologias ou *guidelines* e demais abordagens da ES para um projeto de *software*.

Sendo este *Kernel* uma base para *Softwares* em geral, é possível então estender sua aplicação para áreas de conhecimento específico, sendo objeto deste trabalho a apresentação de uma extensão do *Kernel Essence* para a indústria de jogos digitais. Para isto, no decorrer do artigo iremos apresentar o *Kernel* original e as adaptações realizadas nesta pesquisa para contemplar o que há de elementar para projetos de jogos digitais.

Este trabalho contribui ao dar visibilidade para o *Essence*, visto que ainda é bastante recente sua homologação junto ao *Object Management Group* (OMG) [SEMAT 2018], carecendo de mais estudos sobre o uso deste na modelagem de abordagens de GP em projetos de jogos digitais.

Este trabalho está organizado como segue. Na Seção 2, descrevemos os principais conceitos associados a este estudo. Na Seção 3, apresentamos trabalhos correlatos, sendo outros levantamentos na área de jogos. Na Seção 4, dissertamos e discutimos os resultados da coleta de dados. Por fim, na Seção 5, relatamos nossas considerações finais e trabalhos futuros da pesquisa.

2. Referencial Teórico

Esta seção apresenta aspectos essenciais para compreender o *Essence*, o *Kernel*, os artefatos que o compõe e características da indústria criativa de jogos digitais, a qual estamos caracterizando para nossa extensão deste *Kernel Essence*.

2.1. Indústria Criativa de Jogos Digitais

Segundo a definição proposta pela [UNESCO 2013], a indústria criativa é um termo aplicado que inclui bens e serviços produzidos pelas indústrias culturais e que dependem da inovação, incluindo muitos tipos de Pesquisa & Desenvolvimento (P&D). Eles asseveram que a indústria criativa é bastante abrangente, compreendendo desde a literatura, música, performance artística e artes visuais em seu núcleo, até anúncios, arquitetura, *design* e

moda em sua expressão mais abstrata. Entre as formas mais amplas de indústria criativa, estão os jogos para console e computadores.

A indústria criativa de jogos digitais é uma das principais fontes de renda mundial [Wijman 2022]. Em 2022, o faturamento do setor ultrapassou 184 bilhões de dólares, com queda de 4.3% em relação a 2021. Os jogos web e para computadores obtiveram aumento em suas vendas, já consoles e celulares registraram perda de receita. Essa demanda diversificada abrange ampla gama de categorias de jogos, incluindo simuladores, esportes, ação, aventura, RPG e jogos educacionais. Os projetos de jogos variam em escopo, tamanho e orçamento, desde produções independentes até grandes jogos AAA, desenvolvidos por empresas globais com equipes especializadas [Cruz 2016]. O GP de jogos digitais é desafiador devido a sua natureza interdisciplinar e à necessidade de inovação para se destacar no mercado.

2.2. *Essence*

Com o propósito de estruturar as bases comuns da ES, o SEMAT foi constituído por um grupo de profissionais para formalização de uma base que representa tudo que, comumente, é visto em projetos de *software*. O *Essence* é formado de um núcleo base (*Kernel*) e uma linguagem formal para modelagem das etapas, atividades e abordagens do projeto. De acordo com [Jacobson et al. 2012], o *Kernel* é uma estrutura de prática esquelética, a partir da qual muitas práticas diferentes podem ser usadas em conjunto com facilidade. Ele pode ser estendido para uso em diversos domínios e adaptado conforme as regras de negócio a serem aplicadas. A linguagem *Essence* é formada por um conjunto de símbolos e textos que possuem sintaxe e semântica própria. Os artefatos do *Kernel* utilizam estes para sua modelagem. Tanto o *Kernel* quanto a linguagem estão presentes em [SEMAT 2018], publicado pela OMG.

2.2.1. *Kernel Essence*

Dentre os artefatos que compõem o *Essence*, o *Kernel Essence* está relacionado aos **Alfas (Alpha)**, **Estados Alfas (Alpha State)**, **Espaços de Atividade (Activity Spaces)** e **Competências (Competency)**, representados na Figura 1. Estes artefatos fornecem itens abstratos e essenciais para a organização e rastreabilidade do que se deve fazer e saber em qualquer empreendimento de ES. Cada um dos elementos que compõem os artefatos do *Essence* é inserido dentro de uma das três **Áreas de Interesse (Areas of Concern)** com foco nos principais aspectos envolvidos na ES. (i) **Clientes**: representa tudo o que tem a ver com o uso e exploração reais do sistema de *software* a ser produzido; (ii) **Soluções**: compreende tudo necessário para realizar a especificação e desenvolvimento do sistema de *software*; (iii) **Esforços**: representa tudo relativo à equipe e à maneira como o trabalho é realizado.

2.2.2. *Alfas*

Os **Alfas (Alphas)** capturam os principais conceitos envolvidos na ES, permitindo que o progresso e a integridade de qualquer empreendimento de ES sejam rastreados e avaliados, fornecendo o terreno comum para a definição de métodos e práticas da ES. Por

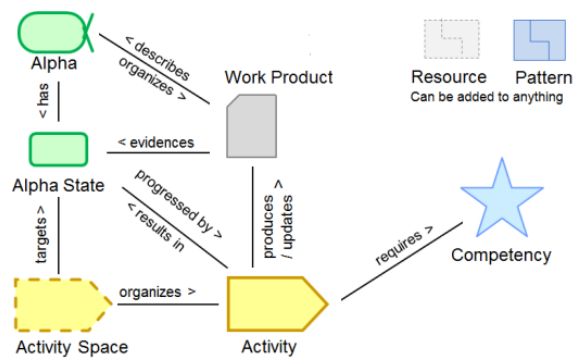


Figure 1. Visão Geral dos Elementos do *Kernel Essence* [SEMAT 2018]

definição, o *Kernel* apresenta sete (7) Alfas comuns que estão presentes em qualquer projeto de software e que estão vinculados a cada uma das áreas de interesse. (i) **Parte Interessada**: São as pessoas, grupos ou organizações que afetam ou são afetados por um sistema de software; (ii) **Oportunidade**: Conjunto de circunstâncias que torna apropriado o desenvolvimento ou alteração de um sistema de software; (iii) **Requisitos**: O que o sistema de software deve fazer para abordar a oportunidade e satisfazer as partes interessadas; (iv) **Sistema de Software**: Um sistema composto de software, hardware e dados que fornece seu valor principal pela execução do software; (v) **Trabalho**: Atividades que envolvem esforço mental ou físico, realizadas para atingir um resultado; (vi) **Equipe**: Grupo de pessoas responsáveis pelo desenvolvimento, manutenção, entrega ou suporte de um sistema de software; (vii) **Método de Trabalho**: O conjunto personalizado de práticas e ferramentas usadas pela equipe para organizar e realizar seu trabalho.

2.2.3. Estados Alfas

Cada Alfa se relaciona através de ações com outros Alfas, permitindo então identificar seu escopo de abrangência dentro do projeto. Além disso, cada Alfa possui um conjunto de estados que permitem identificar e rastrear sua posição atual e futura dentro do projeto. Cada Alfa é representado na forma de um cartão, no qual há uma descrição textual da sua definição e cada um de seus estados dentro do projeto para concluir com êxito sua participação. Cada um dos **Estados Alfas** (*Alpha States*) deriva em um novo cartão contendo uma lista de verificação (*checklist*) das atividades que determinam o estado. A situação do Alfa no atual estado do projeto é validado conforme este conclui os itens do *checklist* de atividades.

2.2.4. Espaço de Atividades

O **Espaço de Atividades** (*Activity Spaces*) é determinado pelo *Kernel*, o qual complementa os **Alfas** provendo uma base visual das atividades. Na área de interesse **Cliente**, a equipe deve entender as oportunidades e envolver as partes interessadas: (i) Explorar as Possibilidades; (ii) Entender as Necessidades das Partes Interessadas; (iii) Garantir a Satisfação das Partes Interessadas; (iv) Usar o Sistema. A equipe deve desenvolver uma **Solução** (área de interesse) apropriada para explorar as oportunidades e satisfazer

as partes interessadas: (i) Entender o Sistema; (ii) Modelar o Sistema; (iii) Implementar o Sistema; (iv) Testar o Sistema; (v) Implantar o Sistema; (vi) Operar o Sistema. Na área de interesse **Esforço**, a equipe deve ser formada e desenvolver o trabalho de acordo com o método de trabalho acordado: (i) Preparar o Trabalho a Fazer; (ii) Coordenar as Atividades; (iii) Apoiar a Equipe; (iv) Monitorar a Progressão; (v) Encerrar o Trabalho.

2.2.5. Competências

O *Kernel* define **Competências** (*Competencies*) para complementar Alfes e Espaços de Atividades, fornecendo as capacidades necessárias para o trabalho no projeto. O número de competências não precisa coincidir com o número de Alfes, mas ambos devem representar as pessoas envolvidas nas atividades. Os artefatos no *Kernel* são adaptáveis, pois as competências variam de projeto para projeto. Em geral, o *Kernel* inclui as seguintes competências. (i) Representação das Partes Interessadas; (ii) Análise; (iii) Desenvolvimento; (iv) Teste; (v) Liderança; (vi) Gerenciamento.

3. Trabalhos Relacionados

[Pieper 2015] introduz uma abordagem integrada baseada em jogos para ensinar métodos de ES. Este trabalho é baseado na especificação SEMAT Essence, que consolida todas as dimensões essenciais dos empreendimentos de ES em um *Kernel* compacto, universal e acionável. A abordagem de jogo integrada ajuda os alunos a entender os conceitos *Essence* de forma mais eficaz. Este trabalho pode ser considerado como um trabalho relacionado em estudos futuros sobre métodos de ensino de ES.

No estudo de [Kemell et al. 2018], os autores apresentam um jogo de tabuleiro inovador que tem como objetivo ensinar a teoria da essência SEMAT a futuros gerentes de projetos e engenheiros de software. O jogo utiliza elementos de gamificação para envolver os jogadores em uma experiência prática e divertida, permitindo-lhes explorar conceitos-chave da ES. Os autores destacam os benefícios potenciais do uso da gamificação no ensino de conceitos da ES, incluindo maior engajamento dos alunos e melhor compreensão dos princípios fundamentais. Este estudo é relevante para pesquisas relacionadas ao ensino de engenharia de software e pode fornecer *insights* valiosos para o desenvolvimento de abordagens educacionais inovadoras nessa área.

[Zapata-Jaramillo et al. 2020] propõem um jogo de tabuleiro interativo para ensinar os diferentes elementos do padrão SEMAT Essence, que é utilizado no processo de desenvolvimento de software. O jogo é baseado em práticas e atividades relacionadas ao padrão e pode ser adaptado para outras áreas, como *Scrum*, *Unified Process*, *Continuous Delivery* e *Automated Testing*. O estudo incluiu testes com estudantes universitários, que mostraram resultados positivos em relação à aprendizagem dos conceitos do padrão. A Tabela 1 sumariza os trabalhos relacionados comparando-os com este estudo.

4. Extensão do Kernel para Jogos Digitais

Nesta seção, apresentamos o trabalho realizado na extensão do *Kernel Essence* para modelagem de abordagens de ES para JDs. Vamos complementar com mais esclarecimentos e elementos gráficos para melhorar a compreensão dos conceitos apresentados na Seção 2.2.1. Este trabalho foi realizado nas etapas iniciais da pesquisa de mestrado do autor

Table 1. Trabalhos Relacionados do Essence com Jogos.

Artigos	Nossa Pesquisa	[Pieper 2015]	[Kemell et al. 2018]	[Zapata-Jaramillo et al. 2020]
Publicado em	2023	2015	2018	2020
Escopo	Extensão do <i>Kernel</i>	Ensino do <i>Essence</i>	Gamificar o ensino de ES	Gamificar o ensino de ES
Contribuições	Integração do <i>Kernel</i> à indústria de jogos digitais	Questionamento da aplicação do <i>Essence</i> no ensino gamificado da ES	-	<i>Boardgame</i> de simulação para ensino da modelagem com <i>Essence</i> .

com uso do *feeling*, buscas *ad hoc* de termos, perfis profissionais e processos encontrados na literatura cinza e a documentação oficial da especificação do *Essence*, desejando apresentar o potencial latente do *Essence* para a indústria de JDs.

4.1. Kernel Essence Estendido

Os artefatos base e seus elementos do *Kernel Essence* (Seção 2.2) tiveram alterações não apenas de simples tradução, mas adaptações coerentes à realidade comum presente em projetos de jogos digitais. Inicialmente, as áreas de interesse tiveram seus termos adaptados para Jogador, antes Cliente, e Jogo, antes Solução, como vemos na Figura 2.

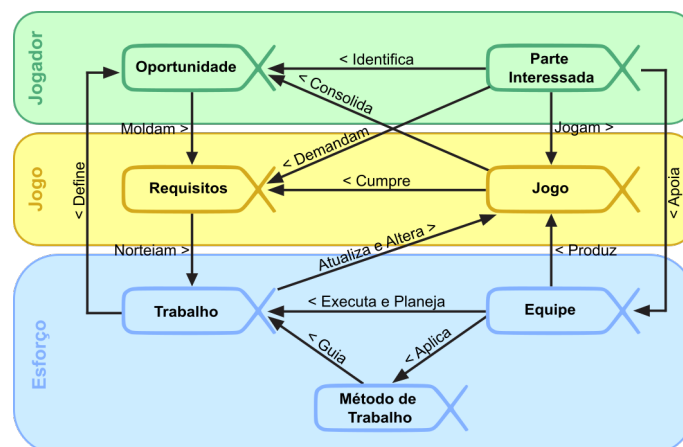


Figure 2. Alfes do Essence para JDs, adaptado de [Jacobson et al. 2012].

4.1.1. Alfes

Entre os **Alfas** do *Essence*, podemos identificar (Figura 2) os mesmos elementos que compõem o padrão adotado pelo *Kernel*, mas com uma modificação: a alteração do **Sistema de Software** para **Jogo**. Esta modificação se justifica porque é o objetivo fim de qualquer projeto de JD. (i) **Oportunidade**: Representa o conjunto de circunstâncias identificadas pela parte interessada para o desenvolvimento ou manutenção do jogo; (ii) **Parte Interessada**: São as pessoas, grupos ou organizações que afetam ou são afetados pelo projeto do jogo, podendo exemplificarmos como *Leads* externos ou fornecedores de tecnologia, em caso de JDs para *console*; (iii) **Requisitos**: É o conjunto de demandas fornecidas pela parte interessada, o qual o jogo deve cumprir com foco nas oportunidades de negócio identificadas; (iv) **Jogo**: É o conjunto de software, hardware e dados usados pela parte interessada e representa a finalidade precípua do projeto. Auxilia na abordagem das oportunidades identificadas pelas partes interessadas; (v) **Trabalho**: Atividades que

envolvem esforço mental ou físico, realizadas pela equipe que executa e planeja estas atividades. O trabalho é relativo ao escopo e restrições impostas pelos requisitos e visa atualizar e alterar o jogo. Segue um fluxo circular, definindo abordagens que podem gerar novas oportunidades e, subsequentemente, novos requisitos; (vi) **Equipe**: Grupo de pessoas, apoiadas pela parte interessada, responsáveis por conduzir a produção do jogo; (vii) **Método de Trabalho**: Conjunto personalizado de práticas, artefatos e ferramentas aplicados pela equipe para guiar o trabalho, integrando os membros da equipe no planejamento e condução do mesmo.

4.1.2. Estados Alfa

Para cada **Alfa** que compõe o *Kernel Essence*, estipulamos o conjunto de estados e criamos um **Cartão** correspondente à descrição dos seus estados, conforme mostra a Figura 3, que representa o estado do **Alfa Jogo**. Os **Estados do Alfa** são sequenciais, iniciando pelo mais ao topo. No caso da Figura 3, iniciaria o estado do jogo a partir da Plataforma, seguindo por Demonstração, Utilização, Pronto, Publicação e Descontinuidade. Para identificar no GP quais **Estados Alfa** já foram alcançados, até o momento da condução do processo, e quais ainda precisam ser concluídos, foram elaborados cartões de Listas de Verificação (*checklist*) para cada estado (Figura 5).

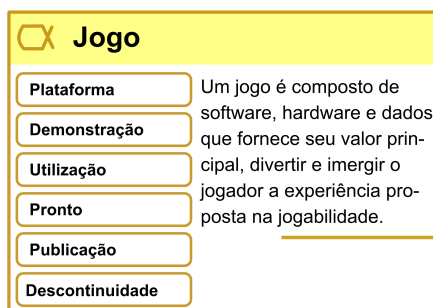


Figure 3. Estado Alfa Jogo

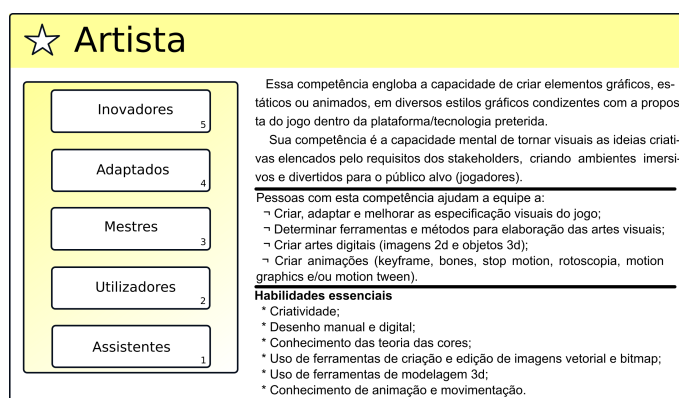


Figure 4. Cartão de Competência

Estes cartões apresentam no seu topo o nome do **Alfa**, abaixo, em destaque, o nome do estado e, só então, o conjunto de itens, em formato de Caixas de Marcação (*checkbox*) com o conjunto de atividades a serem realizadas para a conclusão deste estado. Além destes cartões, para melhor compreender a nomenclatura adotada na formação dos estados e colaborar na identificação de cada item do **Estado Alfa**, foi construída uma **Descrição Completa das Listas de Verificações (Full Checklist)** para cada estado dos **Alfas**, conforme podemos observar no exemplo apresentado na Tabela 2.

Na **Descrição Completa da Lista de Verificação**, a primeira linha apresenta o nome do estado e uma descrição do que ele representa. Abaixo, fica cada um dos itens da **Lista de Verificação** correspondente àquele estado, juntamente com breve texto descritivo do objetivo a ser alcançado para que possa ser considerado concluído.

Table 2. Descrição da Lista de Verificação do Estado Plataforma do Alfa Jogo

Plataforma: Análise para definições iniciais do perfil de hardware, suas características, potencialidades e limitações.	
Critério de Seleção	Os critérios a serem usados na seleção da plataforma foram definidos.
Tecnologias Utilizáveis	Softwares, linguagens de programação, extensões de arquivos, características de áudio e imagem aceitáveis pela plataforma escolhida.
Limitações	Limitações conhecidas da plataforma escolhida.
Decisão de Projeto	Definidas as escolhas de tecnologias e as adequações ao projeto, conforme as limitações da plataforma.
Decisões de Compras	As decisões de compra, construção e reutilização foram tomadas.
Principais Riscos	Os principais riscos do projeto na plataforma.

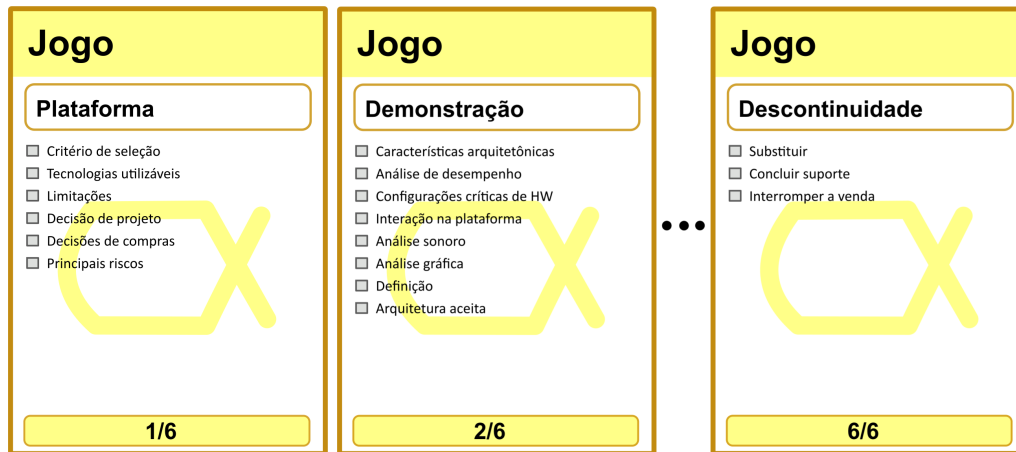


Figure 5. Lista de Verificação dos Estados do Alfa Jogo

4.1.3. Competências

O *Kernel Essence* define também as **Competências** para complementar e dar sentido aos **Alfas** e aos **Espaços de Atividades**, fornecendo as necessidades técnicas para realizar o trabalho durante o projeto. O número de **Competências** não necessariamente coincide com o dos **Alfas**, conforme podemos ver na Figura 6. Porém, estas devem representar as habilidades que cada **Alfa** deve exercer para o sucesso do projeto. (i) **Representante da Parte Interessada**: Capacidade de reunir, comunicar e identificar as oportunidades para outras partes interessadas do projeto; (ii) **Líder de Equipe**: Colabora com a união de tarefas, através da coordenação, colaboração e entendimento entre os demais membros da equipe, das partes interessadas e o gestor do projeto; (iii) **Artista**: Compreende a idealização, execução e avaliação das atividades gráficas de criação, composição e montagem de elementos gráficos 2D e modelagem de objetos 3D fundamentais para a criação do jogo; (iv) **Desenvolvedor**: Compreende a capacidade de pesquisa, concepção e produção de algoritmos responsáveis pela programação que aliará mecânicas, roteiro, projeto dos níveis, elementos artísticos e musicais do jogo; (v) **Músico**: Idealiza, colabora e cria, por meio da composição os elementos sonoros de ambientação, *backing tracks* e efeitos sonoros (*sound effects*) condizentes com a narrativa, roteiro e ambientação do jogo; (vi) **Projetista de Níveis**: idealiza, produz e avalia a partir da narrativa, ambientes desafiadores, unindo os desafios, mecânicas, elementos sonoros e visuais para entreter e imergir os jogadores, que são uma das principais partes interessadas do projeto; (vii) **Roteirista**: encapsula a capacidade de conduzir a narrativa do jogo, gerindo ambientações, cenários, contexto, personagens e diálogos para entreter e integrar o jogador no mundo do jogo. Para isto, deve organizar, sistematizar e elaborar elementos sólidos e inéditos de roteiro

afim de garantir uma experiência agradável e instigante; (viii) **Testador**: avalia, documenta e orienta correções de possíveis inconsistências nos requisitos e o trabalho que precisa ser melhor integrado para o sucesso do jogo; (ix) **Administrador**: gere, planeja, organiza e traça metas e prazos para a organização funcional da empresa, garantindo as metas finais e mantendo os laços de integração com partes interessadas externas financiadoras e/ou comerciais do projeto, caso existam; (x) **Gestor do Projeto**: cria, propõe e conduz as atividades do projeto em conjunto com as partes interessadas. Para isto, deve organizar, planejar e avaliar as ações da equipe, integrando os líderes de equipe na formulação de estratégias para que os requisitos sejam respeitados no trabalho de produção do jogo.



Figure 6. Competências, adaptado de [Jacobson et al. 2012].

Cada **Competência** possui um **Cartões de Competência** conforme demonstramos na Figura 4. Neste cartão apresentamos maiores detalhes, como sua finalidades e habilidades necessárias para um profissional corresponder ao perfil proposto pela competência. Além disso, este cartão apresenta o nível de excelência da competência, em uma escala de nível um (Assistente), onde a competência ainda está em formação, até o nível cinco (Inovador), onde é realizada com *expertise* singular, inédita e de excelência.

4.1.4. Espaços de Atividades

A Figura 7 apresenta os **Espaços de Atividades** (*Activity Spaces*) que correspondem às macro atividades que obrigatoriamente são realizadas pelos alfas, independente do gênero, hardware alvo, público e demais requisitos ou características que o projeto de JD possa ter. A maior parte destas atividades é auto explicativa. Porém, podemos ressaltar que afim de contemplar qualquer tipo de projeto, a macro atividade **Desenhar** corresponde ao ato de produzir artes 2D, modelos 3D ou animações para qualquer uma das opções técnicas do projeto, assim como **Sonorizar** corresponde a compor, gravar, masterizar *sound tracks*, *sound effects* e quaisquer outros elementos sonoros do projeto.

Sendo macro atividades, os elementos que compõem os espaços de atividades devem ser adaptados às características do projeto, sendo esta ação e a modelagem das atividades individuais não previstas no *Kernel Essence* ou nesta extensão, pois ferem o princípio da imparcialidade do *Kernel*. Para tanto, deve-se conhecer a linguagem *Essence*, que apresenta um conjunto maior de símbolos, com sintaxe e semântica textual, que permitirá, assim, instanciar o *Essence* dentro de cada projeto, modelando processos, métodos, atividades e artefatos de *software* utilizados para o GP e sua condução.

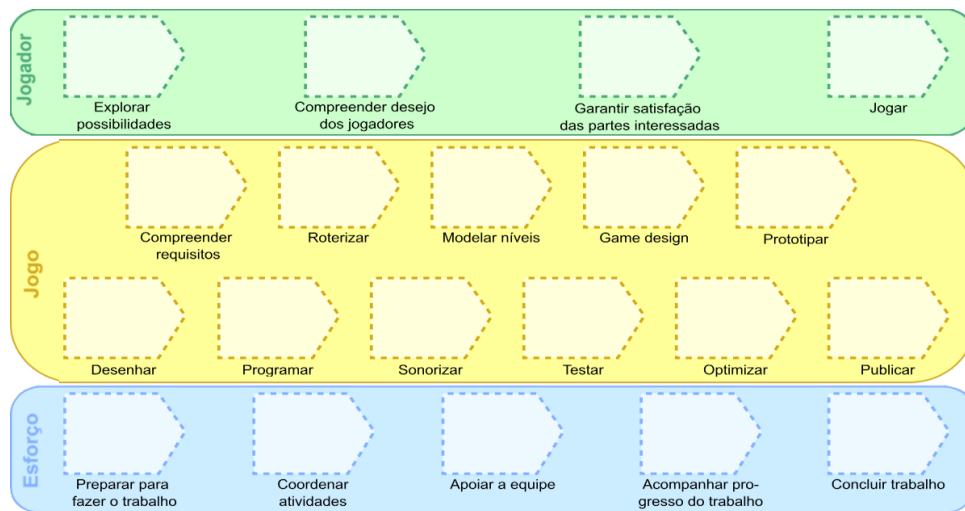


Figure 7. Espaço de Atividade, adaptado de [Jacobson et al. 2012].

5. Considerações Finais

O desenvolvimento de um projeto estruturado de jogos digitais, com base em boas práticas da Engenharia de Software para o gerenciamento de projetos, mitiga eventuais problemas que possam gerar insucesso na Tríplice Restrição (custo, tempo e escopo). Este estudo apresentou uma extensão do *Kernel Essence* para apoiar a modelagem dos elementos essenciais de um projeto de jogos digitais. Através de pesquisas *ad hoc* na literatura cinza, obtivemos um conjunto de informações para compreender os elementos essenciais de qualquer projeto de jogo, independente da abordagem de ES adotada para a condução do gerenciamento de projeto. Este conhecimento norteou o desenvolvimento da extensão proposta para o *Kernel Essence*.

Adaptamos os **Alfas** e seus relacionamentos, personalizamos os **Estados Alfas** e suas listas de verificação, complementamos as **Competências**, criamos **Cartões de Competência** e ampliamos os **Espaços de Atividades** com base nas macro atividades dos projetos de jogos. Dessa forma, construímos a contribuição principal deste trabalho, estendendo o *Kernel Essence* original. Contudo, considerando que este trabalho foi realizado apenas com base na literatura cinza e está, ainda, em fase de avaliação, necessita da realização de uma avaliação para validar sua relevância científica. Com base nos resultados de nossos estudos anteriores, ou seja, uma revisão sistemática da literatura [Pieva and Bernardino 2022] e um *survey*¹ no contexto da indústria de software, poderemos então confrontar este trabalho com a visão acadêmica e da indústria, servindo de base inicial para uma avaliação da proposta do trabalho. Como trabalhos futuros, pretendemos conduzir avaliações experimentais a fim de avaliar o modelo teórico proposto. Entretanto, apesar de ainda não termos avaliado o modelo, estamos confiantes de que estamos em direção ao objetivo proposto.

Agradecimentos

Os autores agradecem à FAPERGS (Projeto 22/2551-0000841-0) pelo apoio ao trabalho.

¹Repositório Zendo com dados do Survey: <https://doi.org/10.5281/zenodo.8103006>

References

- Améndola, F., Fernández, M., and Favre, L. (2015). GLIESE – A Framework for Experimental Game Development. In *2015 12th International Conference on Information Technology - New Generations*, pages 528–533, 2015.
- Cruz, C. A. G. (2016). *Indie Games e a Produção de Jogos no Brasil*.
- Jacobson, I., Ng, P.-W., McMahon, P., Spence, I., and Lidman, S. (2012). The essence of software engineering: The semat kernel: A thinking framework in the form of an actionable kernel. *Queue*, 10(10):40—51.
- Kemell, K.-K., Risku, J., Evensen, A., Abrahamsson, P., Dahl, A. M., Grytten, L. H., Jcdryszek, A., Rostrup, P., and Nguyen-Duc, A. (2018). Gamifying the Escape from the Engineering Method Prison. In *IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pages 1–9.
- Kerzner, H. (2021). *Gerenciamento de projetos: uma abordagem sistêmica para planejamento, programação e controle-2a Edição*. Editora Blucher.
- Kristiadi, D. P., Sudarto, F., Sugiarto, D., Sambera, R., Warnars, H. L. H. S., and Hashimoto, K. (2019). Game development with scrum methodology. In *International Congress on Applied Information Technology (AIT)*, pages 1–6.
- Luhova, T., Blazhko, O., Troianovska, Y., and Riashchenko, O. (2019). The canvas-oriented formalization of the game design processes. In *2019 IEEE 2nd Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, pages 1254–1259, 2019.
- Oliveira, G. W., Julia, S., and Soares Passos, L. M. (2011). Game Modeling using Work-Flow Nets. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 838–843, 2011.
- Pandey, J., Singh, A. V., and Alabri, A. A. (2018). Proposing a hybrid methodology for game development. In *7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pages 142–146.
- Pieper, J. (2015). Discovering the essence of Software Engineering an integrated game-based approach based on the SEMAT Essence specification. In *IEEE Global Engineering Education Conference (EDUCON)*, pages 939–947.
- Pieva, L. and Bernardino, M. (2022). Tendências e Pesquisas em Gerenciamento de Projetos de Jogos Digitais: Uma Revisão Sistemática da Literatura. In *XXI Simpósio Brasileiro de Jogos e Entretenimento Digital*, pages 11–20, Porto Alegre, RS. SBC.
- SEMAT (2018). Essence – kernel and language for software engineering methods, version 1.2. <https://semat.org/documents/20181/57862/formal-18-10-02.pdf/formal-18-10-02.pdf>.
- UNESCO (2013). *Creative Economy Report 2013*. Paris, France.
- Wijman, T. (2022). The games market in 2022: The year in numbers. <https://newzoo.com/insights/articles/the-games-market-in-2022-the-year-in-numbers>.
- Zapata-Jaramillo, C. M., Maturana-Gonzalez, G. V., and Calle-Gallego, J. M. (2020). A Board Game to Simulate the Software Development Process Based on the SEMAT

Essence Standard. In *IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T)*, pages 1–4.