



DESENVOLVIMENTO DO JOGO SUSCITY

Gabriela Trindade Perry
Universidade Federal do Rio Grande do Sul
gabriela.perry@ufrgs.br

Marlise Bock Santos
Universidade Federal do Rio Grande do Sul
marlise.santos@sead.ufrgs.br

Denise Bueno
Universidade Federal do Rio Grande do Sul
denise.bueno@ufrgs.br

Alisson Rocha
Universidade Federal do Rio Grande do Sul
alissonrocha93@gmail.com

Andrei Schwingel
Universidade Federal do Rio Grande do Sul
andrei.schwingel@sead.ufrgs.br

Resumo: Neste artigo relata-se o processo de desenvolvimento do SUSCity, um jogo educacional que tem como objetivo ajudar o jogador a compreender como funciona o Sistema Único de Saúde - e que todos os cidadãos direta ou indiretamente dependem deste sistema. Pretendemos atingir este objetivo usando o sistema de gestão do SUS como mote da mecânica. Sendo assim, o jogador assume o papel de administrador responsável por gerenciar os recursos para tratamento da saúde desta população. Se os recursos forem mal administrados, as consequências do mau planejamento de gestão aparecerão, pois faltarão recursos. O que priorizar? Deve-se priorizar?

Palavras-chave: Projeto de jogos; jogos educacionais; Level Design; Sistema Único de Saúde.

1. INTRODUÇÃO

A associação é direta: se as pessoas gostam de jogos, e se queremos que tenham melhor desempenho e experiências de aprendizagem, devemos dar-lhes jogos educacionais! No entanto, de acordo com O'Neil et al (2005), Linehan et al (2011), e Brom et al (2011), ao mesmo tempo que parece evidente que os jogos têm o potencial para funcionar como ferramentas pedagógicas valiosas, existe pouca evidência de que eles 1) produzem resultados educacionais duradouros, confiáveis, válidos e longos, ou que jogos são 2) mais eficientes do que as estruturas tradicionais de ensino. Ao jogar, as pessoas passam seu tempo tentando alcançar objetivos através de um conjunto bem estruturado de regras. Então, se projetarmos jogos que incorporem as regras desses domínios em sua jogabilidade, talvez os nossos alunos gostassem mais de estudar. Rapini (2012) aponta que os jogos têm sido estudados como ferramentas educacionais por várias organizações internacionais como os projetos *Education Arcade* e *Games-to-Teach*, do MIT; a iniciativa *Serious Games* da Fundação Woodrow Wilson; a Fundação Bill e Melinda Gate.

1.2 Projeto de jogos educacionais

O número de estudos que tentam responder à questão "como projetar jogos" usando o método científico (e publicados em periódicos de renome internacional) é baixo. Entre os exemplos desta abordagem à questão é o trabalho de Martínez-Torres et al (2011), que usou o método Mapeamento Conceitual (Trochim, 1989) para identificar as variáveis-chave para a concepção de ambientes de e-learning. Este trabalho usa um método científico robusto, mas os resultados são um tanto amplos e abertos; não são facilmente aplicáveis ou traduzíveis para guidelines para projeto de jogos. Na mesma direção, Klopfer e Squire (2008) listam cinco propriedades de dispositivos móveis que produzam affordances educacionais exclusivas: portabilidade; interatividade social; sensibilidade de contexto; conectividade e individualidade. Este artigo também traz orientações muito amplas e que não apontam direções claras ao projeto. Mayer (2001) listou os famosos 12 princípios de aprendizagem multimídia, que tem sido muito influentes. No entanto, talvez hoje em dia, eles possam parecer um tanto óbvios - como, por exemplo, o princípio 7, "Contiguidade Temporal", que afirma que "as pessoas aprendem melhor quando palavras e imagens correspondentes são apresentados simultaneamente em vez de sucessivamente". Pensamos que Linehan et al (2011) tenha proposto uma lista mais objetiva de princípios de design que acreditamos que seja mais útil. Suas diretrizes organizadas em 4 categorias, a saber: Definição e comportamento de medição; Gravação e análise de mudança de comportamento; Apresentação de feedback; Adaptação dinâmica para o desempenho dos alunos.

Por este motivo, recomendações sobre projetos de jogos podem ser mais facilmente encontradas em livros do que em artigos científicos. Um excelente exemplo é o livro "Padrões de Game Design" (Björk e Holopainen, 2005), uma coleção abrangente de mais de 200 padrões de design - descrições de elementos recorrentes de interação de jogabilidade que são compartilhadas por vários jogos. Cada padrão tem um nome, uma descrição e é relacionado com outros padrões. Por exemplo: O padrão "Lives" é descrito como "o número de chances que um jogador tem dentro de uma

sessão de jogo antes de ser finalizado" (p.97). De acordo com os autores, os padrões podem ser utilizados tanto para avaliar um jogo como para conceber novos jogos. Por exemplo, o jogo Pac-Man usa os seguintes padrões: "Lives", "High Score List", "Meta Games", "Collections", "Levels", "Role Inversion", "Time limits", "Power-ups" e "Inaccessible Areas" (p.43). Como um exemplo de geração de ideias através da utilização de padrões, os autores sugerem pegar um pequeno número de padrões de forma aleatória, e tentando construir um jogo em cima deles. Este exercício é viável porque os padrões têm relações de instanciação e modulação. Os mesmos autores também desenvolverão um livro similar, porém com foco em jogos para plataformas móveis (Davidsson, Peitz e Björk, 2004).

Como Klopfer e Squire (2008) apontam, jogos educacionais têm que ser orientados ao conteúdo, pois "é o conteúdo é o que nos faz atravessar a porta [da sala de aula]". Linehan et al (2011) destacam a importância do conteúdo com um tom diferente: o conteúdo é certamente importante, mas você não deve se concentrar em demasia sobre isso em detrimento da jogabilidade; ele tem que ser intrínseco à jogabilidade. Um exemplo fictício de má ancoragem para o conteúdo: um jogo onde você tem que salvar uma princesa, respondendo corretamente a problemas algébricos.

2. PROJETO PEDAGÓGICO DO SUSCITY

Em SUSCity apresenta-se uma cidade fictícia como proposta de jogo para aproximar o discente ao Sistema de Saúde. O jogo aproxima o cotidiano do aluno ao tema da saúde na perspectiva da cultura construída por ele antes do conteúdo teórico a ser desenvolvido, e foi construído para contemplar discentes do primeiro semestre do Curso de Farmácia como atividade à distância complementar a disciplina de Saúde Coletiva e Bioética, na qual se debate o sistema de saúde brasileiro no contexto das leis 8080/90 e 8142/90. Nossa visão é que quanto mais as pessoas tiverem o entendimento do significado da participação cidadã na construção da saúde, melhor serão os resultados da mesma. A ideia do jogo é trazer a trajetória de vida no entendimento do aluno, e estimulá-lo a refletir sobre a relação entre saúde, doença e qualidade de vida.

O resultado esperado deste jogo é analisar a construção prévia de conceitos e compará-los aos conteúdos de aprendizagem adquiridos. As implicações esperadas enquanto metodologias participativas são o debate e fixação dos conteúdos disciplinares do processo saúde/doença utilizando o meio virtual, próximo à realidade desta geração. O paradigma deste instrumento de aprender e ensinar é desconstruir o pré-conceito estabelecido por alguns meios midiáticos ao SUS, associando ele apenas ao sistema público de saúde, quando na realidade ele é muito mais do que isto.

Estamos projetando um jogo que terá boa jogabilidade e será desafiador - o que tornará a sessão divertida - mas que também fornecerá o subsídio para mostrar como a gestão da saúde é complexa. O embasamento teórico são as Leis 8080/90 e 8142/90. A proposta do jogo é mostrar a saúde como direito cidadão. Queremos mostrar que o SUS é uma organização do processo de saúde no território nacional. Como o objetivo de SUSCity é promover o debate sobre a importância do SUS e da gestão de recursos para a saúde, enquadrámos o jogo como "casual", ou seja, é caracterizado por missões rápidas, que não exigem que o jogador passe horas jogando (Fortugno, 2008). Além disso, consideramos que, sendo a primeira experiência com

desenvolvimento de jogos desta equipe, seria importante manter a complexidade do projeto sob controle. No caso do SUSCity o delineamento começou com a demanda da primeira autora ao NAPEAD | UFRGS (Núcleo de Apoio Pedagógico à EaD) - promover o debate sobre o SUS - e com o ajustamento das expectativas da professora com as possibilidades tecnológicas. No primeiro contato, a intenção da professora era incrementar e melhorar uma versão anterior do jogo, na qual o jogador construía uma cidade através do posicionamento de prédios, e em seguida, deveria indicar onde tratar os cidadãos. Sugerimos a construção de uma cidade na qual as pessoas estivessem caminhando na rua e buscassem, elas mesmas, pelas unidades de tratamento. A compreensão da mecânica do jogo, por parte da equipe do NAPEAD começou a se delinear quando um dos bolsistas sugeriu que se fizesse "*um tower defense do SUS*". A equipe que desenvolveu o SUSCity pode ser considerada pequena: um designer e um programador, co-autores deste artigo.

2.1 Descrição do jogo

Em SUSCity o jogador é o gestor dos recursos de saúde, que deve decidir como será aplicada a verba em cada cidade. Ele tem a possibilidade de instalar diferentes unidades de saúde, cada uma dedicada a tratar determinados tipos de doenças que as população da cidade pode ter. A verba deve ser aplicada na construção de Unidades de Saúde, que tratarão os pacientes. O SUSCity foi feito inteiramente com o Unity, e possui uma versão para desktop, uma para tablets Android e uma para iPads.

A pontuação em cada nível (cada cidade é um nível) é feita considerando dois parâmetros: a gravidade da doença e o estado final do paciente (se foi curado ou veio a óbito). O gestor precisa decidir como investir a verba em Unidades de Saúde, com o objetivo de atingir as metas de cada nível. Estas metas são expressas em porcentagem mínima de cura e porcentagem máxima de óbitos. Cada cidade contém elementos que informam sobre a riqueza da população, além de edifícios, casas e diversos equipamentos urbanos. Nos espaços vazios, o jogador pode instalar as Unidades de Saúde: UBS (Unidade Básica de Saúde), Farmácia Popular, UPA (Unidade de Pronto Atendimento), Hospital e Emergência. Cidades (níveis) iniciais, que tem dinheiro de sobra para tratar seus doentes, têm muitos equipamentos urbanos, são bem preservados e possuem áreas de lazer (estes elementos denotam que a cidade é rica). A cada nova fase o dinheiro disponível diminui, até que o jogador não consiga mais tratar a quantidade necessária de pessoas, como nas cidades finais, representadas com ruas esburacadas, alagadas e poucos espaços de socialização (ver Figura 1).



Figura 1 - exemplos de cenários

No início de cada nível, o jogador deve decidir em que locais deseja construir as Unidades de Saúde - considerando que cada uma delas tem um custo de construção, uma lotação máxima e um custo de operação, que varia conforme a quantidade e o tipo de doenças tratadas. Antes de iniciar o nível, o jogador sabe quantas pessoas cada cidade tem, o perfil das doenças da população, e o orçamento do Caixa Único da Saúde. É importante ressaltar que todos os pacientes iniciam doentes. O tipo das doenças é refletido na cor da roupa. NPCs (*Non-Playable Characters*) vestindo branco estão curados - todos os demais estão doentes. O tratamento custa tempo e dinheiro. Quanto mais tempo levar para um NPC encontrar tratamento, mais tempo ele levará para ser curado, tornando o tratamento mais caro. É importante ressaltar que os pacientes só saem das unidades de saúde quando estão curados, e que quando um paciente passa em frente a uma unidade, ele entra para ser tratado. Os pacientes não pagam pelo tratamento.

2.2 Processo de desenvolvimento do jogo

O processo de desenvolvimento do SUSCity começou em julho de 2014, mas começou a tomar forma apenas quando o programador entrou na equipe do NAPEAD. Antes disso, tínhamos a ideia da mecânica e algumas artes, mas nada que pudesse ser considerado um jogo.

Em agosto de 2014 tivemos a ideia que deu forma ao jogo, e que foi considerada o marco do início do projeto. A descrição da mecânica foi reportada em nosso sistema de gerenciamento de projetos, e, devido à sua importância fundamental, a inserimos integralmente.

Pensando aqui, eu acho que ficaria mais divertido um jogo no estilo Tower Defense. Seria basicamente uns dois ou três mapas simples, com alguns caminhos (no caso ruas). A ideia é que aos poucos comecem a aparecer do canto da tela unidades inimigas (no caso, pessoas doentes), e o objetivo é não deixar que elas cheguem no fim do mapa e sumam da tela.

Para impedir que as pessoas passem, a ideia é ir criando prédios no caminho (pertinho da rua) e esses prédios geram unidades de defesa (no caso, médicos de vários tipos. Cada tipo iria tratar um paciente diferente). Depois de tratados, os pacientes ficam bem e continuam o caminho pela rua até o fim. O jogador começaria com um certo limite de dinheiro, e toda vez que uma pessoa é curada, ganha-se um pouco mais de dinheiro que pode ser investido em diferentes prédios para tratar mais doentes de maior variedade.

Acho que os outros tipos de prédios (casas, prefeitura, centro da cidade, universidade, etc) poderiam ser construídos para melhorar a efetividade dos médicos, ou poderiam ser feitos como objetivo de chegada dos curados, enfim, teria bastante variedade de possibilidades para utilização.

Os prédios principais (hospital, farmácia e unidade de saúde) seriam os mais usados para combater os doentes e criar unidades. Ou então apenas os hospitais criam médicos, e as farmácias atraem os doentes e eles entram e saem curados (funcionaria só pra doenças leves). Enfim, é pra mostrar que tem várias possibilidades.

Obviamente o jogo não segue fielmente este roteiro – fizemos várias adaptações, como descrevemos anteriormente. Mas tudo começou com esta mensagem.

Em outubro de 2014, a programadora à época estava querendo usar uma engine javascript chamada Solpeo, pois, por demanda da coordenação da equipe, ela havia sido instruída a escolher uma engine. O motivo é que códigos para jogos são extremamente complexos, porém são muito repetitivos. Dependendo do tipo e da plataforma, há certos códigos que precisam ser implementados. Alguns exemplos são: controles de câmera, de path-finding, de colisões, de gerenciamento de sprites, de gerenciamento de cenários, de transição entre animações. Obviamente, os códigos não são os mesmos para um jogo 2D e 3D, mas há, entre os jogos pertencentes a estas duas grandes classes de jogos, muita semelhança. Uma engine cuida destas implementações, deixando o programador responsável apenas pelos códigos do jogo. Em 2016, havia duas grandes engines gratuitas no mercado: a Unity e a Unreal. Nenhuma delas é aberta - e ambas são pagas em alguns cenários. No entanto, a programadora decidiu usar a Solpeo, e deixamos que ela escolhesse. No entanto, a implementação nunca ocorreu. No final de 2014 a programadora foi instruída aprender Unity, mas também não houve sucesso com esta abordagem, e a bolsista foi retirada do grupo.

No início de 2015 (ainda sem o programador atual) decidimos participar do edital Inovapps, e para isso desenvolvemos boa parte das artes para o jogo. Porém não fomos selecionados. Desde então, as artes não sofreram muitas mudanças. Algumas coisas foram incorporadas, e outras foram abandonadas, mas o "*look and feel*" se manteve o mesmo (figura 2).



Figura 2 – Evolução da interface de 2014 até 2016.

Na metade de 2015 o programador se juntou à equipe, e começamos o desenvolvimento efetivamente. Seguindo a sugestão de um experiente professor do curso de Jogos Digitais da Feevale, decidimos modelar o jogo no Excel. Identificamos variáveis importantes e como elas interagiriam juntas. Desta forma, seria possível programar o jogo e fazer os testes de balanceamento com mais foco. Houve algumas mudanças, por exemplo: no início, modelou-se cada doença como tendo uma probabilidade de cura. Nesse sistema, a cada segundo, uma função decidiria se o paciente fica mais ou menos doente. A intenção era fazer com que o progresso da doença não fosse linear (porque na realidade não é), e que pudesse ser controlado pela gravidade da doença. Mas percebemos que isso traria uma complexidade no

código e na análise que não seria traduzida em um aumento na qualidade do jogo. Aliás, várias de nossas decisões relacionadas ao Game Design foram pautadas por esta lógica: se não for melhorar a experiência, não se implementa a função. Por este motivo acabamos simplificando muitas coisas – o mais importante foi o cálculo do orçamento do Caixa Único. Queríamos chegar numa função que calculasse o saldo do Caixa para cada fase, de forma que ele fosse o confortável nas primeiras fases (para não desmotivar o jogador) até que se tornasse progressivamente menor até ficar insuficiente. Ao invés de buscar uma fórmula, usou-se o método da “força bruta”, ou seja, jogamos os mesmos níveis várias vezes, até descobrir as formas de vencer utilizando o mínimo de recursos. Da mesma forma, os valores finais dos custos de tratamento forma definidos através de *playtests* com a equipe de desenvolvimento. Porém, as variáveis e suas correlações estavam definidas internamente.

Esta etapa levou cerca de três meses – com código sendo produzido ao mesmo tempo em que a planilha de balanceamento era criada. Paralelamente, as artes continuavam sendo produzidas.

Depois desta etapa, a programação levou cerca de seis meses para ser finalizada, mais três meses para testes e ajustes. Neste período, foram feitos vários pedidos à arte – a imensa maioria foram ajustes em menus e produção de animações. Nos cerca de três meses de refinamento da programação, executamos alguns testes de usabilidade, que geraram demandas para a arte e para a programação.

Uma das últimas etapas foi a escolha de elementos sonoros – talvez por nossa pouca familiaridade com este tipo de produção, acabamos deixando por último. Buscamos efeitos e loops gratuitos e com licença aberta, pois fizemos modificações em alguns destes arquivos.

2.3 Avaliações de usabilidade

O Suscity passou por 4 avaliações de usabilidade. A primeira, em novembro de 2015, contou com a participação de 3 mestrandos do programa de Pós Graduação em Comunicação da UFRGS. Eles jogaram o jogo em suas casas, e deram *feedback* em uma entrevista com os desenvolvedores. Os pontos críticos que detectamos foram (em ordem de prioridade):

- ✂ A relação entre as doenças e as unidades de atenção não foi percebida.
- ✂ Não ficou claro porque a verba diminui com o passar do tempo.
- ✂ Crença de que havia uma relação entre os edifícios do cenário e as unidades de saúde.

Após esta avaliação, aplicamos um novo código de cores, e passamos usá-lo em diversos pontos da interface do jogo. A Figura 3 mostra algumas das diferenças na interface antes e após a revisão motivada pelo teste.



Figura 3 – Exemplos de alterações após teste de usabilidade.

Para testar novamente o jogo, pedimos ajuda aos professores responsáveis pelo Laboratório de Jogos da Feevale, que gentilmente conduziram um playtest com 5 alunos, no final de fevereiro de 2016. Encontramos bugs relacionados ao áudio, bem como erros de escrita. Não foram encontrados problemas, e consideraram que o jogo estava interessante e pronto para ser distribuído.

Os últimos testes foram realizados durante o primeiro semestre de 2016; em meados de abril e maio em meados de maio de 2016 - ambos com uma turma de disciplina de Saúde Coletiva e Bioética, que tinha 36 estudantes do 1º ano do curso de Farmácia da UFRGS, sob responsabilidade da professora Denise Bueno, uma das coautoras deste trabalho. Infelizmente, no primeiro destes dois testes descobrimos um erro introduzido por uma modificação no código das ambulâncias (os carros que saem do prédio da Emergência e vão atrás dos NPCs doentes) que travava o jogo quando se passava de fase. Estes objetos (computacionais) não estavam sendo adequadamente “destruídos”, de forma que continuavam ocupando memória no jogo – o que eventualmente travava o jogo. Mesmo assim, os estudantes gostaram muito da atividade, e ficaram envolvidos com ela durante 45 minutos (mesmo que o jogo tivesse que ser reiniciado diversas vezes em função do bug).

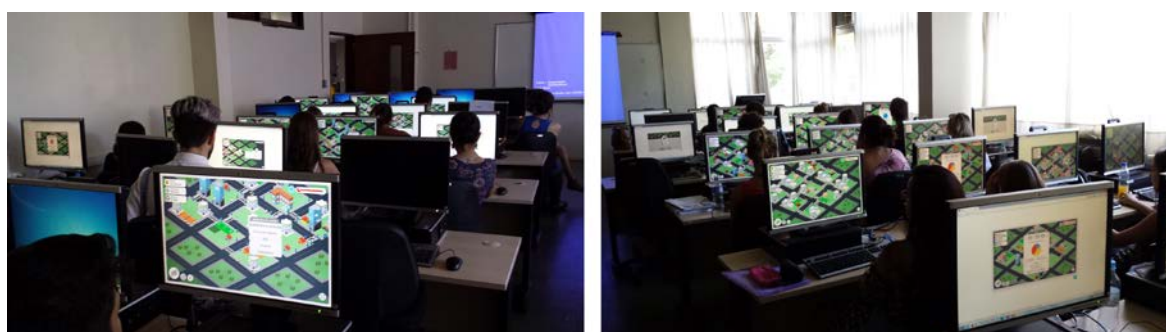


Figura 4 – Avaliação de usabilidade do jogo. Duas perspectivas da sala de informática.

Corrigimos o bug e realizamos o último teste de usabilidade com outra turma da mesma disciplina, no início de maio de 2016. Nesta turma não houve maiores problemas (o jogo travou em 3 computadores).

3 CONCLUSÃO

Consideramos que a abordagem escolhida para o projeto e o desenvolvimento do jogo foram acertadas. Decidimos avançar com cautela, modelando aspectos chave do sistema de saúde com o objetivo de manter a complexidade do jogo sob controle. Esta decisão foi tomada porque além de não ter experiência com projeto de jogos, a equipe considerou que havia uma possibilidade de o jogador não compreender a interação entre as variáveis do ambiente, o que resultaria na não antecipação dos resultados do jogo, o que fatalmente levaria à frustração e ao abandono do jogo.

É importante ressaltar que somos testemunhas de quanto um programador com conhecimento e responsabilidade é essencial em uma equipe como de desenvolvimento de jogos. No nosso caso, que funcionamos mais ou menos como um estúdio de produção de conteúdo formado apenas por bolsistas, é muito difícil encontrar pessoal nesta área. O nosso problema parece ser que o valor da bolsa, e a necessidade de cumprir 20 horas presencialmente não são atraentes para os estudantes que têm conhecimentos em programação. No caso deste estudante em particular, o que lhe atraiu foi a possibilidade de criar um portfólio de programação de jogos - e por esse motivo nós o colocamos para trabalhar apenas com isso. Provavelmente o fato de não termos um professor da graduação em Computação na da equipe aumenta a distância com os potenciais bons estudantes, já que os meios usuais de divulgação das bolsas às vezes não chegam em todos os candidatos na mesma hora.

No caso do designer, o workflow com o programador foi muito facilitado porque o designer estava motivado a aprender Unity. Além disso, o designer já tinha um conceito muito claro para o jogo, e havia encontrado um estilo gráfico que julgamos perfeitamente adequado à visão que tínhamos do jogo. Motivação e destreza com as ferramentas gráficas também são pontos importantes para explicar o desempenho do designer, já que depois de passado o "encantamento" com as etapas mais interessantes da produção visual do jogo (conceito, identidade visual e produção das artes principais) o projeto se torna repetitivo, pois os pedidos são pequenos e relacionados a detalhes - muda a cor dos NPCs, diminui o tamanho do edifício, desenha uma cerca nas casas, troca a cor do botão de instalar unidade, são alguns exemplos dos muitos, muitos pedidos feitos ao designer. Em um dado momento, a equipe de coordenação até mesmo se sentia um pouco constrangida de fazer pedidos ao designer: "olha, [nome do designer], desculpa, mas tem que fazer uns prints do Suscity com um tablet para a PlayStore".

Também consideramos que o desenvolvimento orientado pelo Level Design foi muito acertado, mesmo que os valores das variáveis ainda estejam sendo ajustados. Esta opção permitiu que fossem identificados os componentes chave do sistema, e que a interação entre eles fosse explicitada antes da programação.

Obviamente, consideramos os resultados dos testes de usabilidade fundamentais para o sucesso do jogo, uma vez que as metáforas utilizadas não estavam claras quando terminamos o primeiro build público.

Por último, acreditamos que a qualidade do jogo enquanto produto educacional reside também no casamento entre a mecânica (adaptação de *tower defense*) com o propósito (debater o SUS). Desta forma, conseguimos fazer com que ambos se integrassem harmoniosamente, de forma que o jogo possa ser considerado educacional, sem ser considerado chato.

AGRADECIMENTOS

Os autores gostariam de agradecer à SEAD UFRGS - Secretaria de Educação a Distância - ao NAPEAD – Núcleo de Apoio Pedagógico a EaD – ambos da UFRGS pelo assessoramento pedagógico. Também agradecemos ao Ministério da Comunicação, que através do edital Inovapps 2015 apoiou a realização deste projeto.

REFERÊNCIAS

- BJÖRK, S.; HOLOPAINEN, J. (2005). **Patterns in Game Design**. Charles River Media.
- BROM, C., PREUSS, M., KLEMENT, D. (2011). **Are educational computer micro-games engaging and effective for knowledge acquisition at high-schools? A quasi-experimental study**. Computers and Education, 57(3), 1971-1988.
- CLARK, R. E., FELDON, D. F. (2005). **Five common but questionable principles of multimedia learning**. In R. E. Mayer (Ed.), Cambridge Handbook of Multimedia Learning (pp. 1-22). Cambridge University Press.
- CLARK, R. E. (1994). **Media will never influence learning**. Educational Technology Research and Development.
- CONNOLLY, T.M. (2012). **A systematic literature review of empirical evidence on computer games and serious games**. Computers e Education, 59 (2), p.661-686.
- DAVIDSSON, O., PEITZ, J.; BJÖRK, S. (2004). **Game design patterns for mobile games**. Project Rep. Nokia Research Center, Finland.
- FORTUGNO, N. (2008). **The Strange Case of the Casual Gamer**. IN Isbister, K. e Schafer, N (Ed.) Game Usability: Advance from the experts for advancing the player experience. Burlington: Morgan Kaufmann Publishers.
- GEE, J. P.; MORGRIDGE, T. (2007). **Being a Lion and Being a Soldier Learning and Games**. In. Facer (Ed.), Savannah: Mobile Gaming and Learning? (pp. 1027-1041).
- GEE, J. P. (2008). **Cats and Portals: Video Games, Learning, and Play**. American Journal of Play, 1(2), 229-245.
- GEE, J. P. (2009). **Deep Learning Properties of Good Digital Games: How Far Can They Go?** Theories and Mechanisms: Serious Games for Learning (pp. 63-80).
- KLOPFER, E. SQUIRE, K. (2008). **Environmental Detectives – the development of an augmented reality platform for environmental simulations**. Educational Technology Research and Development, 56, 203-228.

- KOZMA, R. B. (1994). **Will Media Influence Learning? Reframing the Debate.** Educational Technology, 42, 7-19.
- LI, M.-C.; TSAI, C.-C. (2013). **Game-based learning in science education: a review of relevant research.** Journal of Science Education Technology, 22, 877-898.
- LINEHAN, C.; KIRMAN, B.; LAWSON, S.; CHAN, G. G. (2011). **Practical, Appropriate, Empirically-Validated Guidelines for Designing Educational Games.** CHI 2011. Vancouver, Canada.
- MARTÍNEZ TORRES, M.R.; TORAL, S.L.; BARRERO, F. (2011). **Identification of the design variables of eLearning tools.** Interacting with Computers, 23 (3), pp. 279-288.
- Mayer, R. (2001). **Multimedia Learning.** Cambridge Press.
- O'NEIL, H. F., WAINESS, R., BAKER, E. L. (2005). **Classification of learning outcomes: evidence from the computer games literature.** Curriculum Journal, 16(4), 455-474.
- RAPINI, S. (2012). **Beyond textbooks and lectures: Digital game-based learning in STEM subjects.** McLean, Virginia: Center for Excellence in Education.
- TROCHIM, W. M. K. (1989). **An introduction to concept mapping for planning and evaluation.** Evaluation and Program Planning, 12(1), 1-16.