

Um Método para o Desenvolvimento de Software com Crianças Utilizando o Ambiente Scratch

Stephanie da Silva Ribeiro, Amanda Meincke Melo

Universidade Federal do Pampa (Unipampa) – Campus Alegrete
Av. Tiarajú, 810 – Ibirapuitã – 97.546-550 – Alegrete – RS – Brasil

sstephanie.ribeiro@gmail.com, amanda.melo@unipampa.edu.br

Resumo. Crianças podem atuar como coautoras de tecnologias digitais de seu interesse. Ambientes de programação visual, como o Scratch, utilizados para o desenvolvimento do pensamento computacional e lógico no ambiente escolar, são poderosos aliados nesse processo. Como resultado de uma abordagem qualitativa de pesquisa, conduzida a partir de dois estudos exploratórios, este artigo apresenta uma alternativa de integração entre Interação Humano-Computador e Engenharia de Software no desenvolvimento de tecnologia para/com/por crianças, de forma sistematizada, com o apoio do ambiente Scratch.

Abstract. Children can act as coauthors of digital technologies of their interest. Visual programming environments, such as Scratch, used for the development of computational and logical thinking in the school environment, are powerful allies in this process. As a result of a qualitative research approach, based on two exploratory studies, this article presents an alternative of integration between Human-Computer Interaction and Software Engineering in the development of technology for/with/by children, in a systematized way, with the Support environment Scratch.

1. Introdução

Tecnologias digitais são usadas pervasivamente em todo o mundo e devem ser desenvolvidas tendo em vista as experiências e as necessidades de seus usuários finais. O Design Participativo (DP) tem proporcionado a usuários a participação direta durante todo o desenvolvimento de tecnologia para seu uso. Através dessa abordagem, aliada a métodos e práticas de Interação Humano-Computador (IHC) e de Engenharia de Software (ES), a criança, como usuária de tecnologias digitais, pode participar ativamente durante o ciclo de desenvolvimento de *software*, tornando-se coautora direta. Essa coautoria é potencializada com a adoção de ambientes visuais de programação, como o Scratch [Scratch Brasil 2014], com os quais crianças podem gerar produtos de *software*, no todo ou em parte.

Na análise de trabalhos relacionados da Comunidade Brasileira de Informática na Educação, percebeu-se que o ambiente Scratch tem sido utilizado prioritariamente para o desenvolvimento do pensamento computacional e lógico, sem haver necessariamente a preocupação com o desenvolvimento sistemático de um produto de *software* com crianças. Numa perspectiva de Engenharia de Software, entretanto, o desenvolvimento de *software* transcende o processo de codificação e, assim, possibilita

pensar em alternativas para o processo de coautoria de tecnologias digitais por crianças, além de contribuir à reprodução da experiência. O objetivo deste trabalho, portanto, é apresentar uma alternativa de integração entre IHC e ES no desenvolvimento de tecnologia para/com/por crianças, de forma sistematizada, com o apoio do ambiente Scratch. Uma abordagem qualitativa de pesquisa foi adotada, conduzindo-se dois estudos exploratórios.

O texto está organizado como segue. Na Seção 2, é realizada a revisão de literatura, na qual é apresentada a síntese dos trabalhos relacionados. Na Seção 3, é apresentada a metodologia de pesquisa, onde os dois estudos realizados são descritos. Na Seção 4, são apresentados os resultados e as discussões. Por fim, na Seção 5, são realizadas as considerações finais deste trabalho.

2. Revisão de Literatura

A partir de pesquisas nas bases do Simpósio Brasileiro de Informática na Educação (SBIE), do Workshop de Informática na Escola (WIE) e da Revista Brasileira de Informática na Educação (RBIE) por trabalhos que abordam o uso do ambiente Scratch, selecionaram-se aqueles que sugeriam no título a ênfase no produto e/ou no processo de *software*, descartando-se aqueles que sugeriam seu uso apenas como ferramenta de desenvolvimento do pensamento computacional.

Scaico *et al.* (2012) apresentam a experiência de introdução do ensino de programação em uma escola pública de Ensino Médio, com uma abordagem que busca o desenvolvimento da capacidade cognitiva e de resolução de problemas nos alunos. A metodologia envolveu o desenvolvimento de jogos simples e animações, adotando alguns princípios de uma abordagem de programação baseada em *design* (concepção, personalização, colaboração e reflexão), que enfatiza conhecimentos necessários para a criação de um *software* que contemple os interesses pessoais do jovem para a elaboração dos algoritmos.

Sobreira *et al.* (2013) relatam o uso de Tecnologias Digitais de Informação e Comunicação (TDIC) para a introdução da programação e do pensamento computacional, voltada ao desenvolvimento do cidadão do século XXI. Defendem que a ‘Alfabetização Digital’ é importante para que o indivíduo passe de consumidor de tecnologia para desenvolvedor de tecnologia. O Scratch, então, foi usado como ferramenta auxiliar nesse processo, por ser atraente e de fácil entendimento. Os autores mencionam os benefícios pedagógicos com o uso do Scratch e as possibilidades de desenvolvimento do raciocínio lógico na experimentação e resolução de problemas com o seu uso. Além disso, mostram algumas produções com Scratch, realizadas durante uma ‘Oficina de Jogos Digitais’, mas não descrevem o modelo de processo ou as etapas envolvidas.

França e Amaral (2013) abordam a importância da introdução de conceitos da Computação para que, desde cedo, o pensamento computacional seja desenvolvido. Como ferramenta de auxílio, adotam o Scratch, por ser um ambiente de programação visual. Os autores descrevem os conceitos computacionais explorados no Scratch. O processo usado é adaptativo e o desenvolvimento é iterativo e incremental. Práticas computacionais são mencionadas como teste, depuração e abstração, mas não são descritas as etapas do processo.

Arantes *et al.* (2014) descrevem um projeto que tem como objetivo o desenvolvimento de uma geração tecnologicamente autônoma no qual o Scratch é usado como ferramenta de introdução à programação. A abordagem é desenvolvida em encontros: Apresentação e motivação; Primeiro contato com Scratch; Aula expositiva com noções de algoritmos; Scratch com variáveis e condicionais; Dinâmica ao ar livre; Um jogo com Scratch; Criando seu próprio jogo com Scratch; e Avaliação. Embora a proposta dos autores se aproxime da proposta deste trabalho, seu objetivo se distancia ao buscar apenas a introdução à programação, enquanto que este trabalho se propõe ao desenvolvimento de sistemático de um produto de *software* para/com/por crianças.

Rodriguez *et al.* (2015) falam sobre a importância do pensamento computacional para as crianças e nos benefícios obtidos por esse público: o uso de tecnologias que facilitem o envolvimento no planejamento e na elaboração de um ‘produto’ que tenha significado pessoal para a criança, promovendo o reconhecimento da autoria no ‘produto final’. Descrevem o projeto, mencionando o perfil de participantes (14 a 16 anos) e o planejamento das atividades, que foram desenvolvidas em três etapas: exploração dos recursos do Scratch; planejamento, implementação e avaliação de um “projeto” de Jogo Educativo; e elaboração do Relatório Final. No entanto, os autores não apresentam uma abordagem sistemática que integre métodos e técnicas de IHC e ES.

Os trabalhos pesquisados, portanto, apresentam um olhar educacional sobre a abordagem da programação com o auxílio do ambiente Scratch, mas em nenhum deles há o objetivo explícito de desenvolver um produto final com crianças, mas sim de desenvolver o pensamento computacional e um indivíduo dotado de conhecimentos para produzir tecnologias. Por essa razão, é introduzida a programação aos seus conhecimentos.

3. Metodologia

Foram realizados dois estudos exploratórios de desenvolvimento de tecnologias digitais com crianças, adotando-se a abordagem do Design Participativo (DP). Ambos os estudos foram previamente planejados, levando-se em consideração os ambientes onde seriam aplicados, o tempo disponível, métodos e práticas de Engenharia de Software (ES) e de Interação Humano-Computador (IHC), além de possíveis riscos a serem enfrentados. Para cada estudo, criou-se uma agenda, na qual eram discriminadas as atividades que seriam desenvolvidas a cada encontro, bem como o tempo de duração e o dia previsto para que acontecessem.

3.1. Estudo Exploratório 1

O primeiro estudo foi desenvolvido, entre 08 de março e 29 de abril de 2016, em uma casa de acolhimento provisório para crianças e adolescentes, em Alegrete/RS. Teve como objetivo a experimentação de técnicas de IHC e ES, de modo que essa experiência pudesse orientar o planejamento de um segundo estudo.

As atividades foram organizadas, conduzidas e registradas por 4 membros da equipe do projeto de pesquisa “Design Inclusivo para/com Crianças e Adolescentes de Tecnologias Digitais de Informação e Comunicação” (DICA-TDIC) do Campus Alegrete da Universidade Federal do Pampa (Unipampa). O registro do que aconteceu a

cada encontro era realizado de forma escrita, em diários de campo, e através de registro fotográfico das atividades, com autorização da direção da casa de acolhimento e concordância das próprias crianças.

Participaram, inicialmente, cinco meninas com idades entre 13 e 18 anos. Ao final das atividades, o grupo ficou reduzido a três participantes com idades entre 9 e 13 anos, sendo dois meninos e uma menina.

Num primeiro momento, investigou-se sobre o conhecimento das crianças em informática, realizando-se também a exposição da proposta do estudo e a identificação de interessados em colaborar no estudo. Em seguida, foi proporcionado um contato das crianças com o ambiente Scratch.

Após contato preliminar e exploratório com o ambiente Scratch, de cerca de 2 horas, distribuídas em 3 encontros, seguiu-se à etapa de Engenharia de Requisitos (ER), realizada com o apoio das técnicas *brainstorming* [Carvalho e Chiossi 2001] para a exploração de ideias, *braindrawing* [Muller *et al.* 1997] para a exploração de uma ideia selecionada a partir da técnica *brainstorming*, desenvolvimento de Histórias de Usuário [Lucassen *et al.* 2016] para organização dos requisitos de usuário e *storyboarding* [Muller *et al.* 1997] para representar graficamente algumas situações de uso. Então, realizou-se a etapa de Design Iterativo (DI), utilizando a técnica *mock-ups* [Muller *et al.* 1997] e o próprio ambiente Scratch.

Com o uso do ambiente Scratch, o “Jogo das pessoas humanas” foi desenvolvido. Durante a execução deste estudo, a rotatividade dos participantes foi grande, devido a diversos fatores, como participação em outras atividades, perda de interesse, adoção ou alcance da maioria. Portanto, as crianças que tiveram o contato inicial com o ambiente Scratch, nem sempre estavam presentes nas etapas seguintes, de ER e DI, nas quais ocorreram contribuições de crianças que ingressaram posteriormente. Isso ocasionou com que nem todos os participantes percebessem com clareza as possibilidades de desenvolvimento no ambiente, interferindo nos resultados gerados por eles. Apesar disso, ao final do estudo, as crianças que ainda colaboravam conseguiram identificar suas contribuições no produto gerado.

Durante este primeiro estudo, percebeu-se nas crianças a dificuldade de expressão de ideias na forma escrita, sendo a forma oral e o desenho os meios mais produtivos de obter sua contribuição. Observou-se também a necessidade de um período de introdução ao ambiente Scratch de forma mais sistematizada, para que as crianças conseguissem ter segurança ao propor ideias e desenvolvê-las no ambiente. Ainda, a experiência com a técnica *mock-ups*, durante o Design Iterativo, mostrou-se suficiente para compreender o uso da tecnologia em desenvolvimento.

3.2. Estudo Exploratório 2

O segundo estudo ocorreu, entre os dias 09 de agosto e 16 de novembro de 2016, em uma escola estadual da rede pública de ensino de Alegrete/RS. As atividades, alinhadas ao Programa Mais Educação, eram realizadas no período inverso, pela manhã, com estudantes do Ensino Fundamental. Com este estudo, buscou-se compreender se um contato maior e sistematizado com o ambiente Scratch, anterior às atividades de ER, poderia contribuir para que os envolvidos colaborassem com mais segurança à definição de uma tecnologia e ao desenvolvimento de soluções, passando de colaboradores a

coautores na codificação da tecnologia. A cada encontro, os registros das atividades foram realizados em um diário de campo e com fotografias, com autorização da escola, consentimento dos responsáveis e concordância dos participantes.

O planejamento deste estudo envolveu a organização de um método de desenvolvimento de *software* com auxílio da BPMN (do inglês, *Business Process Model Notation*), contemplando as seguintes etapas: Introdução ao Scratch, Engenharia de Requisitos, Design Iterativo e Validação do Produto. Métodos e práticas foram adaptados para que atendessem melhor às necessidades de expressão e à capacidade de compreensão das crianças envolvidas.

Logo após uma avaliação diagnóstica sobre os conhecimentos das crianças envolvidas sobre informática, oficinas de Scratch foram oferecidas a três grupos de crianças, de terceiro, quarto e quinto anos. Cada grupo realizou sua oficina em 6 encontros de 1 hora cada. Ao final, apenas o grupo com maior interesse e integração continuou a contribuir com o estudo até o final, formado por sete estudantes do terceiro ano (8 a 9 anos). As atividades contaram com o apoio de um segundo pesquisador, atuando no registro fotográfico de alguns encontros, na reprodução em alta fidelidade dos desenhos das crianças e durante o desenvolvimento no ambiente Scratch. Propunha-se, assim, oferecer apoio à reprodução dos desenhos e atuar na resolução de dúvidas que viriam a surgir durante a programação do jogo.

Após contato inicial com o ambiente Scratch, a etapa de ER foi iniciada. Esta envolveu as técnicas de *brainstorming*, *braindrawing* e levantamento de Histórias de Usuário, para que fosse definido o que seria desenvolvido durante o estudo. Através do *brainstorming*, com apoio de desenhos com uma breve descrição em texto, todas as crianças desenvolveram propostas de jogos para serem desenvolvidos no ambiente Scratch. As ideias geradas foram levadas a uma votação e a ideia do jogo “Caça à Gata” foi escolhida. Com apoio da técnica *braindrawing*, desenvolveram-se os cenários do jogo, onde todos puderam contribuir em todas as propostas de cenário. Depois de os cenários serem consolidados, os personagens e os obstáculos utilizados no jogo foram desenhados pelas crianças. Estes também foram definidos através de um processo de votação. Então, o levantamento de Histórias de Usuário foi conduzido com o apoio de um painel (Figura 1), apresentando as questões Quem? O que? e Para que?, além de notas adesivas de três cores, indicando o nível de prioridade (ex.: alto, médio e baixo) de desenvolvimento para cada história. As perguntas eram respondidas em conjunto, através de discussão entre as crianças. Somente após um acordo entre elas e a definição do nível de prioridade de cada história, as respostas às questões eram registradas pela pesquisadora nas notas adesivas e fixadas no painel, de modo que ficassem visíveis a todos os participantes. A partir das Histórias de Usuário, realizou-se a divisão de fases do jogo, definindo-se e validando-se o escopo do produto a ser desenvolvido.



Figura 1. Participação no levantamento de Histórias de Usuário

Para a etapa de Design Iterativo, definiram-se três iterações para a codificação do jogo. Cada iteração correspondia à codificação de uma fase do jogo, com seus desafios e obstáculos. Assim, cada iteração foi descrita, pelas crianças, em um painel, determinando o que gostariam que fosse contemplado. Então, as crianças criaram *mock-ups* dos cenários (Sala/Casa, Rua e Cemitério), do personagem principal, do personagem secundário, comidas e obstáculos. Finalmente, codificou-se iterativamente cada cenário (Figura 2). Nesse momento, a pesquisadora interferiu o mínimo possível, contribuindo principalmente na correção de *bugs*.



Figura 2. Participação na codificação do jogo com o ambiente Scratch

Ao final, o jogo foi validado. Cada criança pôde jogá-lo com o intuito de avaliar se estava de acordo com o que havia sido idealizado. Além disso, as crianças conduziram uma avaliação sobre sua participação durante todo o estudo.

O jogo foi desenvolvido quase que totalmente pelas crianças, sendo necessário o apoio da pesquisadora apenas para que algumas dúvidas fossem sanadas e realizadas as correções de alguns *bugs*. Ao final do estudo, as crianças reconheceram sua autoria durante todo o desenvolvimento do jogo “Caça à Gata”, desde a idealização até sua codificação.

4. Resultados e Discussões

No desenvolvimento do primeiro estudo, observou-se que alguns métodos e práticas adotados pressupunham o domínio da leitura e da escrita, dificultando a participação de algumas crianças e exigindo mediação pelos pesquisadores. Assim, uma abordagem mais visual e oral deveria ser considerada em novas experiências. No segundo estudo foi oferecido um contato sistematizado com o ambiente Scratch para que as crianças tivessem um melhor entendimento daquilo que poderia ser desenvolvido e sua coautoria fosse realizada de forma mais segura. Além disso, os métodos utilizados foram adaptados de modo a valorizar a representação visual e a comunicação oral.

As crianças realizaram contribuições significativas, não apenas no desenvolvimento do produto de *software*. Expressaram suas ideias sobre o andamento do segundo estudo, propondo que a realização do *brainstorming* ocorresse em forma de desenhos, para representar melhor cada ideia, e que houvesse uma reorganização nas etapas na produção de protótipos com auxílio da técnica *mock-ups* e na codificação das iterações, o que levou a decisão de representar graficamente todos os cenários, personagens e demais elementos para, então, serem codificados. Essas intervenções das próprias crianças resultaram na adaptação do método, para que sua produtividade fosse mantida e os resultados expressassem da melhor forma o que idealizaram.

Em ambos os estudos as crianças reconheceram sua autoria no produto final. No segundo estudo, em particular, percebeu-se a evolução das crianças com relação aos

conhecimentos em programação, lógica e resolução de problemas. Além disso, as crianças envolvidas demonstraram interesse em continuar participando no desenvolvimento de tecnologia e de estudos semelhantes, evidenciando um sentimento significativo de coautoria em tudo aquilo o que foi produzido.

Além do desenvolvimento de dois jogos, “Jogo das pessoas humanas” (Figura 3a), desenvolvido no primeiro estudo, e “Caça à Gata” (Figura 3b), desenvolvido no segundo estudo, propõe-se um método de desenvolvimento de tecnologia para/com/por crianças com o apoio do ambiente Scratch, onde são integrados métodos e práticas de IHC e ES. Têm-se, ainda, algumas lições aprendidas.

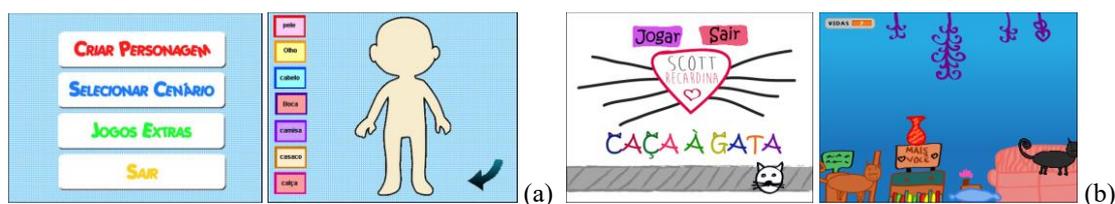


Figura 3. (a) Jogo das pessoas humanas, (b) Caça à Gata

4.2. Método proposto

As experiências vivenciadas nos estudos exploratórios contribuíram à definição de um método de desenvolvimento de *software* por crianças com Scratch, representado em BPMN, na Figura 4, e descrito a seguir.

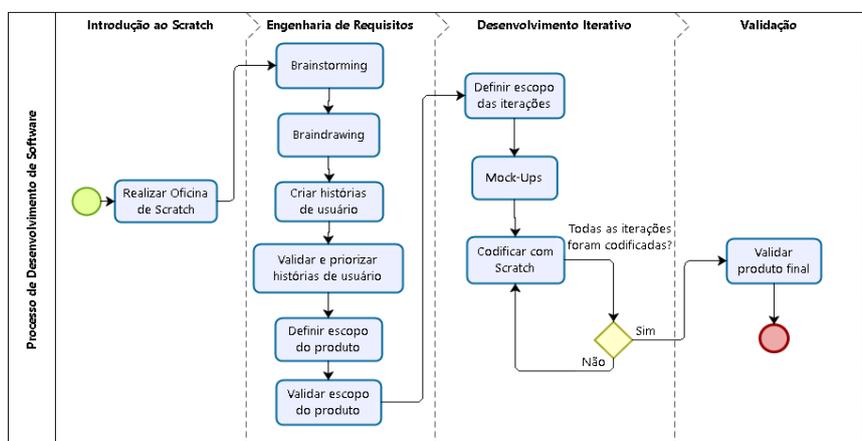


Figura 4. Um método para o desenvolvimento de *software* com crianças utilizando o ambiente Scratch

Na etapa Introdução ao Scratch, é ministrada uma oficina sobre o ambiente Scratch (passo “Realizar Oficina de Scratch”), com o intuito de capacitar as crianças à utilização do ambiente de desenvolvimento. Desse modo, sentem-se seguras a colaborar nas etapas seguintes, na idealização e no desenvolvimento de um produto de *software* no ambiente Scratch.

A etapa “Engenharia de Requisitos” tem início com a realização de um *brainstorming* (passo “Brainstorming”). Essa tempestade de ideias tem por objetivo captar sugestões das crianças de produtos de *software* que possam ser desenvolvidos no ambiente Scratch. Pode ser realizada de forma desenhada, tendo em vista a melhor expressão de ideias pelas crianças.

Então, as crianças exploram a ideia de forma gráfica (passo “Braindrawing”). Para isso, cada criança dá início, por exemplo, ao desenho de um cenário e, a cada 35 segundos, entrega seu desenho para a criança a sua direita. Esse ciclo se repete até que a contribuição de todos, em todas as propostas, seja feita. Essa estratégia pode se repetir para o desenho de personagens, entre outros elementos.

Em seguida, criam Histórias de Usuário (passo “Criar histórias de usuário”) com auxílio de um painel, onde as seguintes perguntas devem ser respondidas pelas crianças: Quem?, O que? e Para que? – ‘Quem?’ é o autor da ação, ‘O que?’ é a ação realizada, e ‘Para que?’ a finalidade da ação no contexto onde é realizada. Esse painel deve ser fixado em um local visível para as crianças e as perguntas respondidas gradualmente, de forma colaborativa e conversada, para que as ideias sejam amadurecidas e, então, escritas no painel, quando o consenso entre elas é estabelecido.

Para a validação e priorização das histórias criadas (passo “Validar e priorizar histórias de usuário”), notas adesivas de três cores diferentes são usadas para auxiliar em sua classificação, por exemplo, amarelo: muito importante; rosa: média importância; e azul: não importante. Para cada história, é realizada a consulta sobre o seu grau de prioridade, indicado pelas crianças, para que, então, seja preenchida na nota adesiva de cor correspondente e fixada no painel.

Finalmente, ainda na etapa de Engenharia de Requisitos, define-se o escopo do produto (passo “Definir escopo do produto”), considerando as ideias propostas e o tempo hábil para a execução da etapa de Desenvolvimento Iterativo. Nesse passo, deve ser esclarecido às crianças que a implementação do produto vai ser realizada, em sua maioria, por elas próprias, de modo que seja definido algo realizável. Assim que o escopo do produto é definido, ele é validado junto às crianças (passo “Validar escopo do produto”), com o intuito de verificar se todas as ideias estão contempladas.

No início da etapa “Desenvolvimento Iterativo”, o escopo das iterações é definido (passo “Definir escopo das iterações”). Para cada iteração, com apoio de painéis, é especificado o que será desenvolvido (ex.: objetivo, cenários, personagens, obstáculos). Então, os painéis são fixados em um local visível para as crianças.

Com auxílio da técnica *mock-ups* (passo “Mock-ups”), cada criança cria um protótipo de baixa fidelidade, em forma de desenhos, para as diferentes iterações. A escolha dos protótipos que serão usados como referência para a codificação de uma iteração é realizada através de uma votação secreta, a fim de evitar constrangimentos entre as crianças. Para isso, cada participante recebe duas cédulas, para que não vote apenas no seu desenho e a votação acabe sempre em empate, que são depositadas em uma urna e posteriormente contadas. O desenvolvimento dos protótipos, de cada iteração a ser codificada, é realizado de forma iterativa para que a produtividade das crianças se mantenha e os resultados representem melhor o que foi idealizado por elas inicialmente. Os protótipos criados pelas crianças são, então, reproduzidos o mais fielmente possível no computador, por uma pessoa com habilidade em desenhos ou pelas próprias crianças, para que a autoria delas seja respeitada.

A codificação com Scratch (passo “Codificar com Scratch”) tem início com a fixação do painel com a definição do ciclo em um local visível para as crianças, assim como a apresentação dos desenhos da iteração reproduzidos no computador, para que

sejam utilizados no ambiente Scratch. As crianças devem ser posicionadas de modo que o engenheiro de *software* consiga visualizar a tela de seus computadores, e elas a tela do engenheiro de *software*, com o intuito de que possíveis dúvidas sejam sanadas rapidamente, garantindo que a implementação no ambiente Scratch seja realizada de forma fluída e facilitada. Isso deve se repetir até que todas as iterações tenham sido codificadas.

Após a implementação de todas as iterações, na etapa “Validação”, o produto final de *software* deve ser validado junto às crianças (passo “Validar produto final”), para que avaliem se este contempla tudo o que, por elas, foi definido.

4.3. Algumas lições aprendidas

Independentemente do ambiente onde o trabalho é realizado, imprevistos e dificuldades estarão presentes. O desenvolvimento de um planejamento prévio, realista e flexível, é de suma importância para que o andamento das atividades ocorra com segurança. É preciso estar disposto a replanejar, sempre que necessário, buscando alternativas para que as atividades se alinhem ao tempo disponível, através de negociação e diálogo.

Nesse processo, ouvir as crianças é de suma importância, quando se está desenvolvendo algo em parceria com elas. Por terem uma percepção diferente a dos adultos, suas contribuições são de grande valor no desenvolvimento de algo de seu interesse. Ao proporcionar um ambiente onde as crianças se sintam à vontade para expressar suas ideias, obtêm-se muitos resultados a partir de sua própria colaboração. A representação visual dos requisitos, como alternativa à escrita, favoreceu a participação das crianças, tanto na definição dos requisitos quanto na compreensão destes.

Proporcionar às crianças uma imersão inicial no uso do ambiente Scratch, de forma mais sistematizada, contribuiu para que tivessem maior segurança para a idealização do produto de *software* desenvolvido com o ambiente. Observou-se também que a presença de cores diferentes para cada categoria de bloco, no ambiente, favoreceu que as crianças com alguma dificuldade de leitura rapidamente conseguissem identificá-las durante as atividades.

5. Considerações Finais

O desenvolvimento de tecnologia com criança exige uma abordagem diferente do que é descrito nos métodos e práticas que são usados com o público adulto. Algumas crianças têm dificuldade em expressar suas ideias de forma escrita, sendo mais fácil o fazer de forma oral e com a representação visual do que está sendo feito, com cores e desenhos.

A introdução, sistematizada, ao ambiente Scratch, no segundo estudo – necessidade identificada durante o primeiro estudo –, trouxe às crianças maior segurança para que atuassem como coautoras. Ao adotar o ambiente Scratch para o desenvolvimento de *software* com crianças, tendo como apoio métodos e práticas integradas de ES e IHC, gera-se um produto de *software* de valor para as crianças e o sentimento de coautoria durante todo o processo de idealização, planejamento e implementação.

Como efeito colateral dos estudos conduzidos, além da capacidade de negociação e de trabalho em grupo, observou-se o desenvolvimento, através da

programação no ambiente Scratch, do raciocínio lógico e do pensamento computacional pelas crianças. A intervenção das crianças, à medida que se sentiam mais confiantes, foi fundamental para o desenho do método.

O método proposto neste trabalho pode ser usado para o desenvolvimento de *software* por crianças com o ambiente Scratch, além de ser possível sua adaptação para a aplicação em outros ambientes de desenvolvimento. A sua adaptação pode ser realizada em trabalhos futuros, assim como a sua aplicação, para análise de sua eficácia em outros cenários de desenvolvimento de *software* para/com/por crianças.

Agradecimentos

À equipe do projeto de pesquisa DICA-TDIC. Ao programa de extensão GEInfoEdu – Grupo de Estudos em Informática na Educação. Ao Programa de Desenvolvimento Acadêmico (PDA 2016) da Unipampa.

Referências

- Arantes, F. L., Amiel, T., Fedel, G. (2014) “Nos rumos da autonomia tecnológica - desafios e lições aprendidas para a formação de jovens”, Anais do Workshop de Informática na Escola, Núcleo de Informática Aplicada à Educação.
- Carvalho, A. M. B. R., Chiossi, T. C. S. (2001) “Introdução à Engenharia de Software” 1. Ed. São Paulo: Editora da UNICAMP.
- França, R. S., Amaral, H. J. C. (2013) “Proposta Metodológica de Ensino e Avaliação para o Desenvolvimento do Pensamento Computacional com o Uso do Scratch” Anais do Workshop de Informática na Escola.
- Lucassen, G., Dalpiaz, F., Van der Werf, J. M. E. M., Brinkkwemper, S. (2016) “Improving agile requirements: the Quality User Story framework and tool” Journal Requirements Engineering.
- Muller, M. J., Haslwanter, J. H., Dayton, T. (1997) “Handbook of Human-Computer Interaction” 2. Ed. Elsevier.
- Rodriguez, C., Zem-Lopes, A. M., Marques, L., Isotani, S. (2015) “Pensamento Computacional: transformando ideias em jogos digitais usando o Scratch” Anais do Workshop de Informática na Escola.
- Scaico, P. D., Lima, A. A., Silva, J. B. B., Azevedo, S., Paiva, L. F., Raposo, E. H. S., Alencar, Y., Mendes, J. P. (2012) “Programação no Ensino Médio: Uma Abordagem de Ensino Orientado ao Design com Scratch” Anais do Workshop de Informática na Escola, Centro de Ciências Aplicadas e Educação.
- Scratch Brasil. (2014) “Sobre o Scratch – Você conhece o Scratch?” Site Scratch Brasil. Disponível em: <<http://www.scratchbrasil.net.br/index.php/sobre-o-scratch.html>>. Acesso em: 29 mar. 2016.
- Sobreira, E. S. R., Takinami, O. K., Santos, V. G. (2013) “Programando, Criando e Inovando com o Scratch: em busca da formação do cidadão do século XXI”, Jornada de Atualização em Informática na Educação.