

PLANO DE TRABALHO PARA O MESTRADO

DESENVOLVIMENTO DE UM MODELO INSTRUCIONAL PARA O ENSINO DE ENGENHARIA DE *SOFTWARE* E USABILIDADE

Universidade Federal de Santa Catarina – UFSC

Programa de Pós-Graduação em Ciência da Computação – PPGCC

Professora Orientadora: Christiane A. Gresse von Wangenheim

1. Introdução

A computação está cada vez mais presente no nosso cotidiano por meio dos diversos dispositivos digitais que estão se tornando indispensáveis para as nossas tarefas. Dessa forma, é essencial que os profissionais do século XXI, independentemente da sua área de conhecimento, tenham uma compreensão dos princípios da Computação. O pensamento computacional, por exemplo, auxilia a análise e desenvolvimento de soluções para os problemas que podem ser resolvidos computacionalmente (CSTA, 2016). Além disso, no Brasil, há uma necessidade de formar profissionais para o setor de *software*, pois existe um déficit na área (CARDOSO e DE DAVID, 2017).

A Sociedade Brasileira de Computação (SBC, 2017) entende que a Computação é uma ciência que deve ser ensinada desde cedo para que no futuro tenhamos recursos humanos qualificados para enfrentar os desafios que advirão. Com esta visão existe a tendência de começar ensinar computação já no ensino Básico. Para este fim, existem diversas unidades instrucionais, por exemplo por meio de programação de jogos ou animações com *Scratch* (OLIVEIRA, 2014), exercícios de programação com blocos (WILSON, 2015) ou robótica (BENITTI, 2009). Uma alternativa é ensinar o desenvolvimento de *apps* em celulares utilizando App Inventor (MIT, 2014). Neste contexto já existem alguns tutoriais (MIT, 2017) e também unidades instrucionais que ensinam fazer um *app*. Estas unidades instrucionais tipicamente enfocam no ensino da programação, não abordando o ensino de conceitos de Engenharia de *Software* (ES) e/ou Engenharia de Usabilidade (EU).

Porém, para ensinar computação de forma mais completa é importante também ensinar competências de ES, como: atividades de análise de requisitos e testes de *software*, como também competências relacionada a EU, como por exemplo, a análise de contexto, prototipação de telas e a realização de testes de usabilidade (CSTA, 2016). A integração destes conceitos no ensino de computação é essencial para assegurar o desenvolvimento de aplicativos com confiabilidade e usabilidade, fatores determinantes de sucesso dos *apps* (PREECE et al., 2005).

Atualmente, o ensino de computação no nível de Ensino Básico foca muito na parte da programação. Além disso, o ensino de conceitos de ES e/ou EU se restringe ao ensino superior. Desta forma, faltam unidades instrucionais que sistematicamente integram estes conceitos de forma apropriado no Ensino Básico.

2. Objetivos

Este trabalho tem como objetivo o desenvolvimento sistemático de um modelo de ensino de ES e EU para o desenvolvimento de aplicativos móveis no Ensino Fundamental 2 com o ambiente de programação App Inventor. O desenvolvimento deste modelo engloba a definição de um processo de desenvolvimento de *apps* integrando sistematicamente práticas de ES e EU, que será ensinado como parte da unidade instrucional.

O objetivo desse trabalho está inserido na linha de pesquisa de Engenharia de *Software* do PPGCC dentro dos tópicos de Processo de Desenvolvimento de *Software* conforme a definição da área de Engenharia de *Software* da SBC (SBC,2005) e alinhado a norma ABNT NBR ISO/IEC 12207:2009.

Objetivos específicos:

O1. Análise da fundamentação teórica sintetizando os conceitos básicos em relação ao ensino de computação no ensino fundamental 2, ao ambiente de programação App Inventor, e à ES e EU.

O2. Levantamento do estado da arte e prática por meio da revisão sistemática de literatura para entender como atualmente os conceitos de ES e EU são ensinados no nível do ensino fundamental.

O3. Evolução de uma unidade instrucional para o ensino de desenvolvimento de aplicativos integrando o ensino de ES e EU.

O3.1. Definição de um processo de ES e EU voltado ao desenvolvimento de aplicativos no contexto do Ensino Fundamental 2.

O3.2. Desenvolvimento de material didático, como por exemplo, slides, roteiros, folhas de tarefas, avaliações.

O3.3. Adaptação/evolução do App Inventor para apoiar o ensino do processo de desenvolvimento de *software*.

O4. Aplicação e avaliação da unidade instrucional desenvolvida em escolas avaliando a unidade em relação à aprendizagem dos alunos, bem como sua efetividade, conforme o modelo de avaliação dTECT (WANGENHEIM et al., 2017).

3. Contribuições científicas potenciais.

Este trabalho tem como principal contribuição científica a elaboração de um modelo de ensino de Engenharia de Software e Engenharia de Usabilidade para o ensino fundamental 2. Além disso, prevê-se as seguintes contribuições científicas:

- Levantamento do estado de arte e prática de forma sistemática fornecendo uma visão geral sobre esta questão de pesquisa;
- Definição de um processo de desenvolvimento de *apps* integrando práticas de ES e EU customizado ao contexto do ensino fundamental 2;
- Desenvolvimento de uma unidade instrucional para introduzir práticas de ES e EU no ensino de computação no nível de ensino fundamental 2;
- Dados e resultados de avaliação sistemática do ensino de ES e EU por meio de estudo de caso;

Uma contribuição tecnológica do presente trabalho será o aprimoramento do ambiente de programação App Inventor para suportar adequadamente o processo definido e o modelo de ensino.

No âmbito social as escolas de ensino básico poderão utilizar a UI desenvolvida, incluindo todo material didático. O trabalho também visa a popularização da computação contribuindo de forma prática à sua aplicação e para o crescimento dessa área de conhecimento. Além disso, contribui com a formação da população em geral, como também estimular o interesse para a atuação nesta área.

4. Possíveis orientadores

Segue abaixo a lista dos possíveis orientadores e linha de pesquisa em ordem de preferência:

Ordem	Orientador	Linha de Pesquisa
1	Christiane Gresse von Wangenheim	Engenharia de <i>Software</i>
2	Patrícia Vilain	Engenharia de <i>Software</i>
3	Carina Friedrich Dorneles	Banco de Dados
4	Ronaldo dos Santos Mello	Banco de Dados

Referências

ABNT NBR ISO/IEC 12207:2009 - **Engenharia de sistemas e software - Processos de ciclo de vida de software**. Disponível em: <http://www.abntcatalogo.com.br/norma.aspx?ID=38643>. Acesso em: Maio 2017.

BENITTI, F. B. V. et al. **Experimentação com Robótica Educativa no Ensino Médio**: ambiente, atividades e resultados. In: Anais do Workshop de Informática na Escola. p. 1811-1820, 2009.

CARDOSO, E.; DE DAVID, T. **A falta de profissionais de tecnologia de informação no mercado de trabalho**. Uma Nova Pedagogia para a Sociedade Futura, p. 697-700, 2017.

CSTA, 2016. **CSTA K–12 Computer Science Standards**. The CSTA Standards Task Force - Revised, ACM, New York/USA.

MIT, APP INVENTOR. Disponível em: <<http://appinventor.mit.edu/explore/about-us.html>>. Acesso em: 20 de abril de 2017.

MIT. **Tutorials for App Inventor**. Disponível em: <<http://appinventor.mit.edu/explore/ai2/tutorials.html>>. Acesso em: Maio/2017

PREECE, J. et al. **Design de interação**: além da interação homem-computador. Porto Alegre: Bookman, 2005.

SBC, 2017. **Plano de Gestão para a SBC Biênio Agosto 2015 – Julho 2017**. Disponível em: <<http://www.sbc.org.br/documentos-da-sbc/send/135-eleicoes/999-plano-de-gestao-para-a-sbc-bienio-agosto-2015-julho-2017>> Acesso em: Maio/2017.

SBC, 2005. **Currículo de Referência da SBC para Cursos de Graduação em Bacharelado em Ciência da Computação e Engenharia de Computação**. Sociedade Brasileira de Computação.

WILSON, C. **Hour of code---a record year for computer science**. ACM Inroads, v. 6, n. 1, p. 22-22, 2015.

WANGENHEIM, C. G. V. et al. **dTECT: Um Modelo para a Avaliação de Unidades Instrucionais para o Ensino de Computação na Educação Básica**. INCoD/GQS.02.2017.P (May/2017).