1. **INTRODUÇÃO**

1.1 CONTEXTUALIZAÇÃO DO PROBLEMA

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. As tecnologias que ela provem são encontradas nos mais diversos pretextos: lares, escolas, empresas, impactando tanto na vidas das pessoas como na economia mundial (ANTONIA, 2012). A maioria das profissões, em maior ou menor grau, também utilizam computadores para cumprir suas funções. Assim, é importante que as pessoas, independentemente da sua área de conhecimento, tenham uma compreensão dos princípios da computação. Esta compreensão possibilita resolver problemas de forma computacional, tornando o profissional mais produtivo e competitivo na sua área (CSTA, 2016).

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. É também fundamental para que os alunos possam ser educados tecnologicamente e possibilitar criar novas tecnologias afim de ajudar as vidas das pessoas (Wangenheim, 2014). Além disso, estimula o interesse das pessoas nesta área, sendo que no Brasil há uma necessidade de formar profissionais para o setor de *software*, pois existe um déficit na área (CARDOSO e DE DAVID, 2017). Porém, alguns empecilhos dificulta o ensino da computação no ensino Básico, como a falta de materiais de estudos e recursos humanos qualificados (INEP, 2010 a 2014). Ademais, no Brasil, o ensino da computação se restringi apenas ao Ensino Superior.

Existem diversos currículos com o objetivo de disseminar o conhecimento da computação no Ensino Básico, dentre eles está o modelo de currículo referencial CSTA/ACM K-12 (CSTA, 2016). O currículo define diretrizes que ensina as principais competências da computação, dentre as principais estão a programação e o pensamento computacional, conhecimentos chaves para resolver problemas por meio dos computadores. (CSTA, 2011).

Visando a popularização da computação nas fases iniciais das escolas, existem diversas iniciativas que desenvolveram ferramentas para auxiliar a programação, por exemplo, Scratch (SCRATCH; MIT, 2013) e App Inventor (App Inventor, MIT,2013). Ainda, 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).

Para ensinar computação de forma mais completa é importante também ensinar competências de ES, como: atividades de análise de requisitos e testes de *software*. A ES é definida como uma sistemática de abordagens quantificáveis para o desenvolvimento, operação, e manutenção de software (IEEE CS, 2014). Ainda, é necessário o ensino das 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). Apesar dessas duas áreas de conhecimento divergirem no foco durante desenvolvimento do software, elas definem modelos de processos, métodos e técnicas para o desenvolvimento de software interativo e sistemático (Gibertoni, 2016).

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. Desta maneira, este trabalho pretende desenvolver um modelo para ensinar o desenvolvimento de aplicativos integrando o ensino da ES e EU no contexto do Ensino Fundamental 2. Isto é feito por meio de uma UI, no qual irá conter material didático, como por exemplo, slides, roteiros, folhas de tarefas, avaliações. Esta unidade desenvolvida está alinhado ao currículo de referência ACM/CSTAK-12 (CSTA,2016). Por fim, o trabalho aplica e avalia a UI desenvolvida em escolas avaliando-a em relação à aprendizagem dos alunos, bem como sua efetividade. Para isso, é utilizado o modelo dETECT (Wangeminet, 2017), que visa avaliar sistematicamente a qualidade das UIs focado o ensino da computação Básica com base na percepção dos alunos.

1.2 PERGUNTA DE PESQUISA

A pergunta de pesquisa definida para o presente trabalho é: Como é possível ensinar ES e EU por meio do desenvolvimento de aplicativos móveis focado no ensino fundamental 2 alinhado ao currículo CSTA-K12, aplicando o modelo dETECT para avaliar a qualidade de unidades instrucionais focadas no ensino de computação na Educação Básica com base na percepção dos alunos.

1.3 OBJETIVOS

O objetivo geral e os objetivos específicos são descritos nesta seção.

**Objetivo geral**

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.

**Objetivos específicos**

Os objetivos específicos são:

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*.

04. 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 dETECT (WANGENHEIM et al., 2017).

**DELIMITAÇÕES**

O escopo da UI proposta desenvolvida neste trabalho visa o ensino de desenvolvimento de *apps* para os alunos do Ensino Fundamental 2, isto é, período do 6º ao 9º ano. Este período está relacionado ao modelo de matriz curricular CSTA-K12, mais especificamente ao seu nível 3. A UI limita-se ao desenvolvimento para as plataformas *mobiles* que utilizam Android como sistema operacional. Uma vez que o celular está se tornando comum entre os adolescentes (faixa etária de 10 a 15 anos), visto que 84% dos adolescentes usam o dispositivo (TELECO, 2016). Além disso, é a plataforma dos apps desenvolvido pelo App Inventor (AI), que por sua vez foi escolhido por ser indicado para as pessoas que nunca programaram e desejam desenvolver seus primeiro aplicativos (Wolber, 2012). Portanto, o trabalho não abrange a utilização de outros ambientes, como por exemplo, Strach, Hour of code.

Analisando o estado da arte observa-se que as UIs criadas para o ensino da computação tem tipicamente enfoco na programação. Dessa forma, este trabalho abrange a programação, porém com foco maior no ensino das disciplinas de EU e ES por meio do desenvolvimento de *apps*.

1.4 ADERÊNCIA À CIÊNCIA DA COMPUTAÇÃO

O tema deste trabalho está aderido conforme os objetivos do Programa de Pós-graduação em Ciência da Computação da Universidade Federal de Santa Catarina (PPGCC/UFSC), mais especificamente nos 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.

Ao analisar o regimento interno nº 01/PPGCC/2013, publicado em 01/10/2013, em seu artigo 1º, dentre os objetivos do programa está “o desenvolvimento de novos conhecimentos em Ciência da Computação”. Dentre os conhecimentos que integra a Ciência da computação estão as disciplinas de ES. A ES aborda diversos tópicos, dentre elas a Qualidade de software, que entre os seus atributos que contribuem para a qualidade, está a usabilidade (IEEE CS, 2014). Para o uso sistemático da usabilidade é necessário a Engenharia de Usabilidade, como então sub área de ES.

Neste sentido, este trabalho integra à Ciência da Computação ao desenvolver UIs contendo conhecimentos de ES e EU. Esses conhecimentos são relevantes, pois a UI foi desenvolvida para atender os objetivos educacionais definidos pelo currículo de referência CSTA k-12 voltado para o ensino da Ciência da Computação no ensino básico. Outros conhecimentos relevantes para Ciência da Computação também foram produzidos, tais como, ensino da programação, levantamento atualizado do estado da arte e da prática sobre o ensino de ES e EU no ensino básico e no curso de Ciência da Computação, por meio de uma Revisão Sistemática da Literatura (RSL) e um *Survey*.

Analisando com mais detalhe aos objetivos da linha de pesquisa em ES do PPGCC/UFSC, observa-se a aderência da presente tese a estes objetivos, quais são:

“Engenharia de Software: tem como objetivo formar indivíduos capazes de conduzir o processo de desenvolvimento de software e de investigar novas metodologias, técnicas e ferramentas para a concepção de sistemas.” (PPGCC/UFSC, 2015).

Posto isto, esta dissertação aborda primordialmente o ensino ES e EU no desenvolvimento de apps. Considerado que a ES é parte fundamental do processo de desenvolvimento de software de qualidade, estando presente entre os conhecimentos definidos pelo Software *Engineering Body of Knowledge* (SWEBOK) (IEEE CS, 2014), entende-se que há correlação do tema da tese a este objetivo da linha de pesquisa.

1.5 METODOLOGIA DE PESQUISA

**1.5.1. Contexto de pesquisa e classificação**

O contexto de pesquisa deste trabalho se baseia no método científico em camadas (research-process onion) (SAUDERS, LEWIS, & THORNHILL, 2009), conforme ilustra a Figura 2.



Figura 2. Contexto da pesquisa no método científico (SAUDERS, LEWIS, & THORNHILL, 2009)

O contexto da pesquisa é definido nas seguintes camadas (descritas da mais interna para mais externa): horizonte de tempo, método, estratégia, abordagem e filosofia. Neste sentido, o contexto desta pesquisa é: *Cross-sectional*, pois são analisados os indivíduos, no caso os alunos e os tutores da UI, apenas durante o período de aplicação da UI; Multi-método, pois ao longo desta pesquisa são aplicados diversos métodos, tais como, Revisão Sistemática da Literatura (RSL) (KITCHENHAM, 2004), ADDIE (BRANCH, 2009), desenvolvimento de software ao adaptar a ferramenta App Inventor (STEPHENS, 2015), GQM (BASILI, CALDIERA, & ROMBACH, 1994), etc.; Diversas estratégias são utilizadas, tais como, estudos de caso (YIN, 2014), survey (KASUNIC, 39 2005), observação, e pesquisa de arquivo; Indutiva, pois serão analisados estudos de caso particulares de aplicação da UI, para inferir conclusões gerais sofre seus efeitos. Outro aspecto que caracteriza a abordagem indutiva é que a pesquisa não parte de uma hipótese pré-estabelecida, mas, procura atingir a solução do problema com base nas conclusões inferidas do objeto estudado, no caso, a aplicação da UI em turmas do ensino fundamental 2; e Interpretativista, pois o objetivo desta pesquisa (desenvolver e avaliar uma UI para o ensino de ES e EU no ensino fundamental 2) depende da interpretação do pesquisador sobre as informações coletadas durante a aplicação da UI.

**1.5.2. Etapas da pesquisa**

Este trabalho separa em 4 etapas de pesquisa para o desenvolvimento, aplicação e avaliação da UI, conforme ilustra a Figura 3.

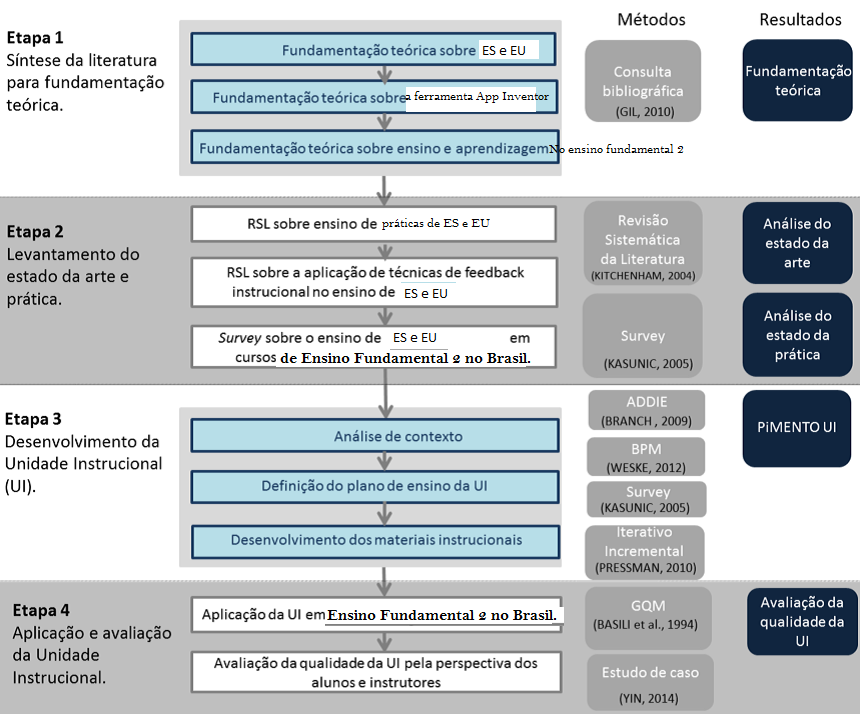


Figura 3. Etapas da pesquisa

O detalhamento do trabalho realizado em cada etapa é detalhado a seguir:

**Etapa 1** - Síntese da literatura para fundamentação teórica

O objetivo desta etapa é fazer um estudo das literatura que abordam atividades das disciplinas de ES e EU. Também é feito uma análise teórica sobre computação móvel e desenvolvimento de aplicativo no Ensino Básico por meio da ferramenta App Inventor. Por fim é feito um estudo teórico sobre o currículo de referência CSTA k-12, no qual será integrado o currículo do ensino Fundamental 2.

**Etapa 2** - Levantamento do estado da arte e prática

Nesta etapa será feito o levantamento do estado da arte para entender como o ensino de ES e EU vem sendo realizado e avaliado no ensino. A análise do estado da arte é realizada em duas partes: 1) a Revisão Sistemática da Literatura (RSL) (KITCHENHAM, 2004), no qual é feito o planejamento da RSL fazendo o levantamento as palavras chaves do estudo, seleção dos principais trabalhos e estudar os assuntos relacionado ao estudo em questão, e 2) O Survey, que é feito o levantamento de como o ensino da ES e EU está sendo realizado no Ensino Básico. Seguindo o processo de Kasunic (KASUNIC, 2005), o survey também tem a etapa de planejamento, no qual é definido o objetivo de pesquisa, o público-alvo, planejamento da amostra, definição das questões de análise métricas, projeto do questionário, e teste piloto do questionário. Por fim, é feito a distribuição do questionário e a coleta de dados, no qual vai servir de base para interpretação do resultado.

**Etapa 3** - Desenvolvimento da UI

Para desenvolvimento da UI foi utilizado as fases da abordagem de ADDIE (BRANCH, 2009) para realizar o design Instrucional. Durante a realização dessas fases são definidos as atividades suas atividades pertinentes. Segue as fases abaixo:

1. Análise: é identificado e caracterizado o público alvo após coletar as informações deu um survey sobre o ensino de programação no ensino Básico.
2. Projeto: é definido e sequenciado o conteúdo da UI conforme o que o currículo CSTA K-12 necessita entorno do ensino de ES e EU para o ensino da computação no ensino básico. Nesta fase também serão definido a estratégias de feedback, e como ocorrerá a avaliação da UI.
3. Desenvolvimento: são desenvolvidos as atividade da UI. Tais atividades são: definir um processo para o ensino de ES e EU para o desenvolvimento de aplicativos móveis, desenvolvimento do material didático, como por exemplo, slides, roteiros, folhas de tarefas. Também é feito a implementação da adaptação da ferramenta App Inventor para apoiar o ensino do processo de desenvolvimento de *software*.

**Etapa 4 -** Aplicação e avaliação da UI

Nesta etapa a UI desenvolvida é colocado em prática e feito uma avaliação da sua efetividade. A UI é aplicada em uma escola de ensino fundamental 2 que servirá como estudo de caso. Primeiramente é definido e planejado a aplicação e a avaliação da UI. Após é feito a execução da aplicação da UI e coletado os dados. A avaliação seá feita por meio de instrumentos de medições do modelo de avaliação dETECT (WANGENHEIM et al., 2017).

1.6 CONTRIBUIÇÕES

A realização desta dissertação produz contribuições nos âmbitos científico, tecnológico e social.

**1.6.1 Contribuição no âmbito científico**

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. Este modelo possibilitará os alunos a aplicar os conceitos dessas disciplinas na construção de *apps* com confiabilidade e usabilidade.

Dentre outras contribuições, está o levantamento do estado da arte atual em relação aos trabalhos que já apresentaram propostas de desenvolver uma UI para o ensino da computação no ensino básico. Analisando estas unidades instrucionais, observa-se que todos focam em ensinar os alunos a programar, não inserindo nos seus objetivos ensinar a EU e ES. Diante disso, este trabalho define de um processo de desenvolvimento de *apps* integrando práticas de ES e EU customizado ao contexto do ensino fundamental 2, incluindo materiais instrucionais para introduzir práticas de ES e EU no ensino de computação no nível de ensino fundamental 2 alinhando ao currículo CSTA (2016).

Por fim, com o intuito de avaliar a UI em relação à aprendizagem dos alunos, bem como sua efetividade, este trabalho contribui cientificamente aplicando e avaliando a unidade instrucional desenvolvida em escolas por meio de estudo de caso seguindo o modelo de avaliação dETECT (WANGENHEIM et al., 2017).

**1.6.2 Contribuição no âmbito tecnológico**

Em relação ao âmbito tecnológico a importância deste trabalho é o aprimoramento da ferramenta App Inventor para suportar adequadamente o processo definido e o modelo de ensino. O aprimoramento desta ferramenta permitirá introduzir a EU por meio da evolução de seus componentes visuais deixando amigáveis e com maior variedade.

**1.6.3 Contribuição no âmbito social**

No âmbito social as escolas de ensino básico poderão utilizar a UI desenvolvida, incluindo todo material didático e a versão aprimorada do App Inventor. Dessa forma, os tutores poderão ensinar, além da programação, a ES e EU, tornando o ensino da computação mais completa.

O projeto ainda 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.

ANtonia, 2012. <http://antoniasousaqueirozzz.blogspot.com.br/>

TELECO, 2016, http://www.teleco.com.br/ncel\_usu.asp

WOLBER, D. **AppInventor.org**. 2012. Disponível em: <http://www.appinventor.org/course-in-a-box>. Acesso em 25 jun. 2015.

IEEE CS. (2014). *SWEBOK - guide to the Software Engineering body of knowledge* (3th ed.). USA: IEEE

CHAO, G. Human-computer interaction: **The usability test methods and design principles in the human-computer interface design**. In: Computer Science and Information Technology. Beijing: China. 2009. p. 283 – 285

LIANG L.; DENG X.; WANG Y**. Usability Evaluation Driven by Cooperative Software Description Framework**. In: Computational Sciences and Optimization. 2009. p. 364 – 366.

e Interação Humano-Computador. A usabilidade faz parte de estudo da disciplina Interação Humano-Computador, sendo usado como um termo que realiza avaliação global de grau de utilização na interação humano-computado (CHAO,2009). As atividades de usabilidade são estudadas pela Engenharia de Usabilidade, sendo aplicada no processo de desenvolvimento de software (LIANG;DENG,2009). Enquanto a ES cuida dos aspectos de desenvolvimento de *softwares*, a EU trata de aspectos comportamentais, relacionados à interação humano-computador. Apesar de serem duas disciplinas serem divergentes em relação ao foco, ambas propõem desenvolver softwares interativos de forma sistemática, definindo modelos de processo, métodos e técnicas (SILVA,2004).