

# Validação da Caracterização de Conceitos do Pensamento Computacional em Atividades da Engenharia de Software

Júlia de Avila dos Santos<sup>1</sup>, Simone André da Costa Cavalheiro<sup>1</sup>,  
Luciana Foss<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Computação  
Universidade Federal de Pelotas (UFPEL)

{julia.asantos, simone.costa, lfoss}@inf.ufpel.edu.br

**Abstract.** *Preparing and developing educational projects present a series of challenges, among them, a systematic approach divided into stages that enables execution and leads to the desired outcomes. This study addresses these challenges by drawing inspiration from Software Engineering (SE) principles and leveraging the concepts of Computational Thinking (CT). CT has gained prominence as a problem-solving methodology, meanwhile SE provides the framework for conceptualizing and executing computational projects. Notably, CT principles are inherent in SE processes. This article aims to validate a characterization of CT concepts in SE activities using the Delphi method. By explicitly linking CT to the stages of project development and execution, educators can utilize such activities to foster and assess CT skills while concurrently enhancing proficiency in other domains.*

**Resumo.** *A elaboração e o desenvolvimento de projetos educacionais apresentam uma série de desafios, entre eles, uma abordagem sistemática dividida em etapas que possibilite a execução e leve aos resultados desejados. Este trabalho aborda esses desafios, inspirando-se na área de Engenharia de Software (ES) e fundamentando-se nos conceitos do Pensamento Computacional (PC). O PC ganhou destaque como uma metodologia de resolução de problemas, enquanto a Engenharia de Software fornece uma fundamentação para concepção e execução de projetos computacionais. Identifica-se que os princípios do PC são inerentes aos processos de ES. Este artigo tem como objetivo validar uma caracterização de conceitos do PC em atividades da ES, usando o método Delphi. Ao relacionar explicitamente o PC às etapas de desenvolvimento e execução de projetos, permite-se que educadores utilizem tais atividades para promover e avaliar habilidades do PC, enquanto aprimoram simultaneamente a proficiência em outros domínios.*

## 1. Introdução

A atividade de projetos educacionais tem se apresentado como um processo que provê uma aprendizagem significativa aos estudantes. Ela é entendida como um processo criativo, impulsionada pelas qualidades pessoais dos envolvidos, que permite construir de forma gradual um produto ou matéria com novidade subjetiva ou objetiva [Ignatyeva et al. 2018]. No entanto, diversos obstáculos têm se apresentado na execução deste tipo de atividade, entre os quais a carência de habilidades para solucionar problemas

[Dyah Utami et al. 2023] e a falta de metodologias claras que permitam a sistematização da elaboração destes tipos de projetos [Aidas et al. 2020, OECD 2018].

O Pensamento Computacional (PC) tem se popularizado como uma habilidade fundamental a todos, a qual permite aos estudantes propor e resolver problemas de maneira eficaz. Por sua vez, a Engenharia de Software (ES) é uma área da Computação que define processos para o desenvolvimento de projetos computacionais, cuja aplicação vem demonstrando resultados efetivos e de qualidade. Ela provê uma série de técnicas que auxiliam na concepção, desenvolvimento, criação e avaliação de produtos de software.

Como primeira etapa para a fundamentação de uma abordagem que permita sistematizar a elaboração de projetos educacionais e em conjunto desenvolver habilidades do PC, em [Santos et al. 2022] buscou-se caracterizar os conceitos/técnicas do PC nas atividades da ES. Para validar as caracterizações estabelecidas está sendo utilizado um painel de especialistas, estruturado pelo método Delphi [Dalkey and Helmer 1963]. Neste artigo, apresentam-se os resultados obtidos na primeira rodada de avaliações. A partir disso, uma versão revisada das caracterizações é apresentada.

Vários trabalhos relacionam o PC e a ES. Apiola and Sutinen (2020) propuseram o uso do DSR (*Design Science Research*), uma estrutura para resolução de problemas, para a aprendizagem do PC e da ES, relacionando-os às competências STEM/STEAM. Para validação da proposta foi realizada uma análise de dados, a partir de uma aplicação prática em um grupo de pesquisa. Por sua vez, a pesquisa de Fronza et al. (2016) propõe a metodologia do pensamento visual e do PC para dar suporte ao ensino de engenharia de software. Para analisar a metodologia proposta foi realizada uma experiência de diferentes abordagens em diferentes contextos, entre eles, ensino médio e ensino fundamental. Já em Fronza et al. (2017), é proposto um *framework* que conta com um conjunto de princípios e práticas, baseado em métodos ágeis de engenharia de software, que podem ser mapeados para o desenvolvimento de projetos de programação (que desenvolvem o PC). Como experimento para avaliação da proposta, o *framework* foi aplicado em duas turmas de sexto ano. Saad (2022) também explora o PC para o ensino de ES, utilizando a abordagem de aprendizagem baseada em projetos. A avaliação dos experimentos foi realizada por meio de pré e pós testes, comparados por um teste-t, o qual teve resultado significativo na diferença de conhecimento dos alunos. Embora todos os trabalhos relacionem o PC e a ES, nenhum deles deixa explícito de que forma as habilidades do PC estão associadas às atividades da ES.

O restante do artigo está organizado como segue. Na Seção 2 são apresentados os fundamentos teóricos que embasam este trabalho: os conceitos e técnicas do PC e a proposta de caracterização de conceitos do PC em atividades da ES. Uma visão geral sobre o método adotado para a validação da proposta é relatada na Seção 3. A Seção 4 descreve as discussões e resultados alcançados, expondo as alterações realizadas após as análises das avaliações e sugestões dos especialistas. Por fim, as considerações finais deste trabalho e seus próximos passos estão delineados na Seção 5.

## **2. Fundamentação Teórica**

O PC se fundamenta em conceitos e técnicas oriundos da Ciência da Computação que podem auxiliar na resolução de problemas. Considera-se neste trabalho: *abstração*, como o processo de tornar um artefato mais compreensível através da redu-

ção de detalhes desnecessários [Csizmadia et al. 2015]; *automação*, como a mecanização das soluções (ou de suas partes), permitindo que máquinas nos ajudem a solucionar os problemas [Ribeiro et al. 2020]; *avaliação*, como o processo de identificar se existe uma solução viável, que seja adequada ao propósito e esteja correta [Csizmadia et al. 2015, Ribeiro et al. 2020]; *dados*, conceito que envolve a coleta, a representação e análise de informações [CSTA/ISTEA 2011]; *decomposição*, que consiste em analisar o problema e dividi-lo em subproblemas, os quais são resolvidos independentemente, e cujas soluções são combinadas para construir a solução do problema original; *generalização/reconhecimento de padrões*, como uma técnica que consiste em construir uma solução (algoritmo) mais genérica a partir de outra, permitindo que o novo algoritmo seja utilizado em outros contextos [Ribeiro et al. 2020]; *paralelismo* como uma técnica na qual várias tarefas são planejadas para serem executadas ao mesmo tempo para alcançar um objetivo, a solução do problema ou parte dele [Brennan and Resnick 2012]; *pensamento algorítmico*, que envolve a elaboração e execução de etapas para solucionar um problema ou parte dele [Shute et al. 2017].

## 2.1. Caracterização do PC em Atividades da ES

Esta subseção sintetiza a caracterização do PC nas atividades da ES propostas em [Santos et al. 2022], as quais foram submetidas ao processo de validação. Após a descrição de cada atividade (prefixadas com A), caracteriza-se a relação considerada com o PC (prefixadas com R). Importante destacar que a numeração não indica uma relação de ordem entre elas.

- A01. Formulação e descrição do problema**, identificando os requisitos necessários para sua solução, possíveis restrições e o objetivo a ser alcançado. **R01.** A formulação envolve a descrição *abstrata* do problema, identificando informações e recursos necessários para a sua resolução e resultados esperados. Nesse momento, não é realizada uma descrição de como, mas sim o que deve ser feito para se obter a solução do problema. Nesta atividade, deve-se identificar explicitamente os *dados* envolvidos na resolução do problema, podendo ser agrupados por tipos abstratos (como conjuntos numéricos, equipamentos, textos etc.), quando for o caso.
- A02. Levantamento e derivação de requisitos** obtidos a partir da interação com atores relacionados com o problema. **R02.** Diversas características e restrições relacionadas ao problema são identificadas durante o levantamento de requisitos (coleta de *dados*). A partir disso, os requisitos serão derivados e a técnica de *abstração* pode ser aplicada para selecionar as características e restrições relevantes, descartando aquelas irrelevantes. Os requisitos devem ser suficientemente detalhados de forma que facilite a descrição do problema, evitando excesso de detalhes que possa complexificar essa descrição. Conforme a necessidade, os requisitos podem ser *decompostos* baixando o nível de abstração (e aumentando o nível de detalhamento). O processo de organização de derivação e gerenciamentos dos requisitos pode ser *automatizada* por meio de ferramentas específicas. No caso em que se identifique diferentes fontes para realização do levantamento e derivação de requisitos, estes podem ser agrupados em categorias, que podem ser tratadas de forma *paralela* por diferentes equipes.
- A03. Análise de viabilidade**, incluindo o levantamento de custos (tempo, financeiro etc.) e verificando se os recursos disponíveis são suficientes para se alcançar o

objetivo estabelecido. **R03.** Ao se fazer o levantamento dos custos e recursos disponíveis, estas informações (*dados*) devem ser organizadas de modo a facilitar a *avaliação* da viabilidade de se resolver o problema. O levantamento e análise pode ser *decomposto* em diferentes aspectos (tempo, custo, equipamentos etc.), permitindo a *paralelização* de suas execuções.

**A04. Validação dos requisitos** identificados, observando sua consistência e abrangência, realizando ajustes quando necessário. **R04.** A *avaliação* da correção do levantamento dos requisitos é realizada verificando se não há contradições entre os requisitos estabelecidos e se abrangem todos os aspectos desejados.

**A05. Estruturação da resolução do problema,** identificando os subproblemas e os seus relacionamentos. **R05.** Para se definir a estrutura da resolução do problema, deve-se *decompô-lo* em subproblemas, identificando os componentes que resolvem cada um dos subproblemas. A forma em que esses componentes se relacionam deve ser estabelecida com base na dependência que cada componente tem dos demais, o que determinará a maneira em que as suas soluções serão *compostas*. Podem ser identificados subproblemas que seguem um mesmo *padrão* para que se possa *generalizar* o componente que os resolve.

**A06. Planejamento de execução,** elaborando o cronograma de resolução dos subproblemas, observando as dependências entre eles e registrando os possíveis riscos envolvidos, bem como alocando a equipe de execução para cada etapa discriminada no cronograma. **R06.** A definição do cronograma envolve a identificação da ordem na qual as ações relacionadas as demais atividades serão realizadas e a alocação da equipe para realizar as diferentes ações. Neste planejamento devem ser identificadas ações que podem ser realizadas em *paralelo*, definindo fluxos independentes de execução, destacando pontos de sincronização. O *pensamento algorítmico* deve ser aplicado na especificação desta ordem, levando em conta as dependências entre os subproblemas e as demais atividades.

**A07. Descrição das resoluções dos subproblemas,** identificando objetivos, restrições e possibilidades de reúso. **R07.** Uma descrição *abstrata* (sem detalhamento) da resolução de cada subproblema indicando o seu objetivo e as restrições envolvidas deve ser apresentada. Ainda, podem-se identificar *padrões* entre os objetivos de diferentes subproblemas para *generalizar* as resoluções e permitir o reúso.

**A08. Definição das interfaces** da resolução de cada subproblema, identificando os recursos necessários e resultados esperados de forma precisa para que os demais componentes que o utilizem não precisem conhecer os detalhes de sua resolução. **R08.** Para cada subproblema, é necessário representar com precisão as informações/recursos (*dados*) necessários para a resolução de cada subproblema, bem como os resultados esperados, levando em conta os relacionamentos definidos entre os subproblemas.

**A09. Resolução dos subproblemas** (criação, reúso ou adaptação), descrevendo de forma detalhada e não ambígua as etapas envolvidas na sua resolução. **R09.** Utiliza-se o *pensamento algorítmico* para a elaboração e a descrição detalhada das etapas da resolução de cada subproblema, indicando explicitamente o uso dos recursos para se alcançar o objetivo. Em caso de reúso, quando adaptações forem necessárias, a resolução deve ser descrita com as respectivas modificações. Para as resoluções que devem solucionar mais de um subproblema, a descrição das etapas precisa ser *genérica* para que se aplique a todos os casos identificados.

Pode-se fazer uso de dispositivos computacionais para *automatizar* algumas das resoluções, implementando a descrição detalhada correspondente.

- A10. Integração das soluções** dos subproblemas, fazendo uso das interfaces estabelecidas e obtendo a resolução completa do problema inicial. **R10.** As resoluções dos subproblemas devem ser *compostas* de forma lógica, respeitando as interfaces e a estruturação do problema, considerando as *paralelizações* previamente identificadas. Essa composição deve ser descrita de modo a se obter um *algoritmo* que constitua a solução detalhada do problema original. Se for o caso, a resolução pode ser *automatizada* através da implementação do algoritmo descrito.
- A11. Verificação da resolução do(s) (sub)problema(s)**, de forma independente, analisando a sua descrição detalhada para identificar possíveis falhas com relação aos objetivos e restrições pré-estabelecidos, corrigindo-as quando necessário. **R11.** A resolução do(s) (sub)problema(s) pode(m) ser simulada(s), considerando quando for o caso diferentes possibilidades de recursos de entrada, para *avaliar* a sua correção. Isto é, verificar se, seguindo os passos da resolução com os recursos dados, é possível chegar no objetivo estabelecido. A *abstração* pode ser utilizada para selecionar características relevantes permitindo categorizar os recursos de entrada (*dados*) em classes. Um elemento de cada classe pode ser selecionado para realização das simulações, simplificando o processo de verificação.
- A12. Verificação da resolução completa do problema**, analisando a descrição integrada das resoluções dos seus subproblemas para identificar possíveis falhas com relação ao objetivo e restrições pré-estabelecidas e às interfaces dos subproblemas, corrigindo-as quando necessário. **R12.** As relações estabelecidas com o PC são as mesmas descritas no item anterior, considerando agora o problema como um todo.
- A13. Execução da resolução do problema**, envolvendo os atores relacionados, para identificar a aceitação da solução com relação às expectativas dos atores, readequando-a quando necessário. **R13.** Os passos descritos na resolução do problema são executados em um ambiente real (*pensamento algorítmico*) para sua validação.
- A14. Proposição de extensões/adaptações** à resolução do problema para atender a novos objetivos, que possam surgir após a interação com os atores. **R14.** A execução da resolução pode fazer emergir novos objetivos que devem ser descritos de forma *abstrata* para que extensões e adaptações sejam adicionadas a descrição da solução do problema.

### 3. Metodologia de Validação da Proposta

O processo de validação das relações estabelecidas foi realizado por meio de um painel de especialistas, elaborado a partir do método Delphi, no qual profissionais e pesquisadores das áreas de ES e/ou PC foram convidados a analisar e comentar a proposta. O método Delphi foi utilizado como ferramenta metodológica complementar para estruturar o processo de comunicação em grupo [Linstone and Turoff 2002].

O método envolve a aplicação de questionários a especialistas em rodadas. No intervalo de cada rodada são feitas análises das respostas e o resultado é compilado em novos questionários que, por sua vez, são novamente distribuídos ao grupo [Kayo and Securato 1997]. Novas rodadas são realizadas até que se obtenha, quando possível [Dalkey and Helmer 1963], o mais confiável consenso entre os especialistas. No que

segue, são apresentadas as etapas do método que foram realizadas, bem como uma breve descrição de como foram executadas.

**Escolha do grupo de especialistas e convite para participação:** foram contactados 12 profissionais que atuavam e/ou pesquisavam nas áreas de PC e/ou ES, por meio de um convite que contextualizava a pesquisa e solicitava sua participação - anônima - no processo de validação. No caso de concordância, eles deveriam dar sua anuência, preenchendo o Termo de Consentimento Livre e Esclarecido (TCLE)<sup>1</sup>. Oito profissionais, 6 doutores e 2 mestres, concordaram em participar da pesquisa, dos quais 3 atuam como professores/pesquisadores de ES, 3 são professores/pesquisadores na área do PC e 2 são professores/pesquisadores em ambas as áreas (PC e ES). Todos os participantes concluíram a pós-graduação há pelo menos 3 anos e ainda atuam nas respectivas áreas. Destaca-se que a maioria dos pesquisadores da área do PC tem a formação mais recente (2020), enquanto a maioria dos profissionais que atuam na área de ES concluíram sua formação até 2007. Cabe observar que o pesquisador com formação mais antiga (1996) atua nas duas áreas, tendo começado a pesquisar em PC há aproximadamente 10 anos.

Nielsen e Landauer (1993) [Nielsen and Landauer 1993] afirmam que 5 participantes são suficientes para se obter uma diversidade de respostas e que, e após a quinta participação, os mesmos achados são observados repetidamente. Com isso, não se buscou uma maior quantidade de avaliadores.

**Construção do primeiro questionário:** o questionário de avaliação foi elaborado via formulário on-line (<https://forms.gle/fhkhsP2jd9QA2Uff7>), com 14 seções, cada uma apresentando uma das atividades adaptadas e sua relação com o PC. Para cada item, era solicitado ao respondente seu grau de concordância com respeito à adequação da descrição da atividade e à adequação da relação estabelecida, utilizando para ambos a escala Likert (de 1, discordo totalmente, à 5, concordo totalmente). No final de cada seção, havia uma pergunta em aberto para o respondente adicionar comentários sempre que desejasse. Solicitou-se em particular que, sempre que possível, as discordâncias fossem justificadas. No cabeçalho do formulário foram sintetizados os conceitos de PC considerados no trabalho.

**Envio do primeiro questionário:** após o aceite de participação da pesquisa e o preenchimento do TCLE, o link de acesso ao formulário era enviado por e-mail para o especialista.

**Recebimento das respostas do primeiro questionário:** todos os especialistas que concordaram em participar da pesquisa responderam o formulário de avaliação, totalizando 8 respostas.

**Análise qualitativa e quantitativa das respostas:** após o fechamento do envio de respostas, ocorreram as análises quantitativas e qualitativas dos resultados obtidos, os quais são apresentados na próxima seção. A partir desta etapa, uma nova versão das caracterizações foi elaborada.

#### 4. Discussões e Resultados

Esta seção apresenta as análises dos resultados da pesquisa, inicialmente descrevendo os dados quantitativos e posteriormente os qualitativos. A subseção 4.1 contém a versão

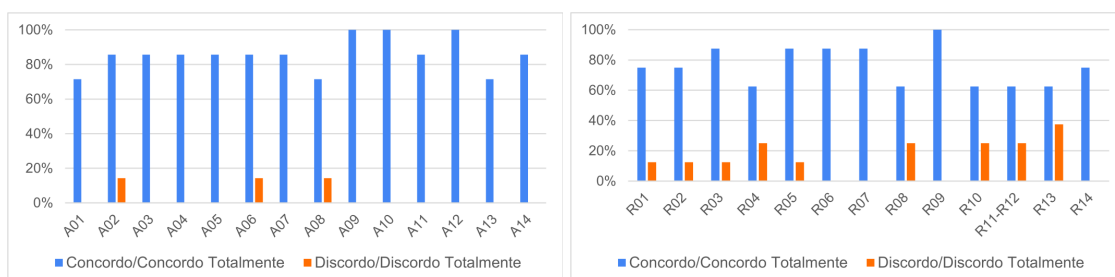
---

<sup>1</sup>TCLE: <https://forms.gle/oZ1m1kujz166Cghm6>. CAAE: 73891417.0.0000.5317.

revisada das atividades adaptadas e das caracterizações que tiveram avaliações negativas ou comentários pontuais negativos.

A Figura 1 apresenta os gráficos que comparam os percentuais de avaliações positivas (concordo e concordo totalmente) com os percentuais das negativas (discordo e discordo totalmente) para cada uma das descrições das atividades (à esquerda) e das relações estabelecidas (à direita). O avaliador 6 foi excluído das estatísticas das descrições das atividades, já que ele mencionou no comentário que não tinha condições de avaliar por não ter familiaridade com as atividades de processos no desenvolvimento de software. Com isso, o percentual referente às atividades foi calculado sob um total de 7 avaliações e o das relações sob um total de 8 avaliações.

Com respeito às avaliações das atividades adaptadas, é possível observar que mais de 80% dos avaliadores concordaram (totalmente) com maioria das descrições. Percebe-se que três atividades tiveram menos de 80% de concordância (A01, A08 e A13) e apenas três delas tiveram avaliações negativas (A02, A06 e A08). Por sua vez, as atividades A09, A10 e A12 tiveram 100% de concordância. Com relação as relações estabelecidas com o PC, a maioria (8 de 13 relações estabelecidas) obteve concordância igual ou superior a 75%. No entanto, cinco tiveram concordância inferior à 75% (R04, R08, R10, R11-R12 e R13). Cabe observar que nenhuma relação teve concordância igual ou inferior à 60%.



**Figura 1. Avaliações positivas versus avaliações negativas das atividades (à esquerda) e das relações (à direita) propostas.**

Os comentários dos especialistas deram subsídios para identificar os motivos das discordâncias, bem como as principais falhas da proposta. De forma geral, alguns termos usados nas descrições não estavam claros ou até mesmo eram inadequados, tornando as redações confusas. Com respeito às relações estabelecidas com o PC, a crítica mais frequente foi que elas não estavam apresentadas de forma explícita. No que segue, elencamos as considerações negativas recebidas para os itens que receberam avaliação positiva inferior a 75%, indicando o que foi realizado nas respectivas reformulações.

- **Considerações A01:** a crítica foi que a definição do problema, tratada nesta etapa, não se relaciona com a solução. Com isso, foi retirada, tanto da descrição da atividade quanto da relação, toda menção à solução ou a qualquer coisa que se referisse a ela.
- **Considerações R04:** observou-se que os termos usados nas descrições não se aplicavam à atividade de validação de requisitos e que a relação com PC não estava explícita. Os termos foram adequados, tanto na atividade quanto na relação, e buscou-se deixar a relação com o PC explícita.
- **Considerações A08 e R08:** as descrições foram consideradas confusas. Também, subentendeu-se que a atividade A08 estaria fora de ordem com relação a resolução

do subproblema. A partir disso, reformulou-se a escrita, tentando torná-las mais compreensíveis. Ademais, a descrição da atividade A07 foi refeita para deixar claro que o subproblema não é resolvido nessa etapa.

- **Considerações R10:** considerou-se que alguns termos usados não estavam claros e que a relação com o PC não estava explícita (ela mais explicava a atividade do que de fato a relacionava com o PC). Assim, procurou-se retirar os termos que poderiam gerar confusão e deixar explícita as relações com o PC.
- **Considerações R11-12:** pontuou-se o uso de termos inadequados na descrição da relação. Os termos foram adequados e as relações com o PC explicitadas.
- **Considerações A13 e R13:** salientou-se que a etapa confunde a execução (implantação) com a validação. Como a intenção era de fato de relacionar com a etapa de implantação, retirou-se a referência à validação.

Cabe destacar que comentários negativos pontuais, tanto para atividades ou relações, também foram considerados para a reformulação da proposta. No que segue, são apresentadas as considerações para os itens que se enquadram neste caso.

- **Considerações A02 e R02:** alguns termos utilizados foram considerados inadequados e a descrição da relação foi considerada implícita e imprecisa. A reescrita considerou a adequação dos termos e a explicitação das relações.
- **Considerações A03 e R03:** a atividade não deixava explícito que ela incluía o levantamento de recursos, além de não considerar possíveis restrições. Outra crítica foi que a análise de viabilidade influencia de forma indireta o problema, não tendo uma relação direta com o PC. A atividade foi reescrita, incluindo explicitamente as ações de levantamento de recursos e restrições, além da verificação das restrições na análise da viabilidade. Também se reformulou e se explicitou as relações com o PC.
- **Considerações R05:** a relação com o PC não estava explícita. Reformulou-se, explicitando a mesma.
- **Considerações A06 e R06:** considerou-se confusa a descrição da atividade. Não houve avaliação negativa para relação, mas a sugestão da adição de mais uma relação com o PC. Adicionou-se a relação sugerida.

#### 4.1. Caracterização do PC em Atividades da ES: versão revisada

Com base nas considerações, uma versão revisada das atividades de processos da ES adaptadas para a resolução de problemas, bem como as respectivas relações com os conceitos do PC foi proposta. No que segue, listamos apenas os itens que sofreram modificações<sup>2</sup>.

**A01. Formulação e descrição do problema**, *identificando e caracterizando o contexto em que o problema se insere (os atores envolvidos, suas capacidades e necessidades), as condições necessárias para a sua resolução e o objetivo a ser alcançado. Nesse momento, não é realizada uma descrição da solução, mas sim o que ela deve abordar.* **R01.** A formulação envolve a descrição abstrata do problema, identificando *as características relevantes da solução e os resultados esperados. Aqui se destaca a relação com o conceito de abstração. Além disso, deve-se descrever explicitamente os dados fundamentais envolvidos no domínio do problema, investigando as relações entre o problema, o objetivo e os dados (ação associada ao conceito de representação de dados).*

---

<sup>2</sup>As alterações realizadas, com relação a proposta original, estão destacadas em itálico.



- A02. Elicitação e derivação de requisitos** obtidos a partir da interação com atores relacionados com o problema. **R02.** *A relação com o conceito de coleta de dados se manifesta na identificação das características e restrições relacionadas ao problema durante a elicitación de requisitos. Ainda, o conceito de abstração se aplica quando são selecionadas as características e restrições relevantes no processo de compreensão dos requisitos. Na etapa de derivação, os requisitos devem ser suficientemente detalhados de forma que facilite a descrição do problema, evitando excesso de detalhes que possa complexificar essa descrição (abstração). Conforme a necessidade, eles podem ser decompostos baixando o nível de abstração (e aumentando o detalhamento). Neste caso, fica expressa a relação com o conceito de decomposição. O processo de derivação e gerenciamentos das especificações dos requisitos pode ser realizado com o suporte de ferramentas específicas. A seleção e o uso dessas ferramentas relacionam esta etapa com o conceito de automação. No caso em que se identifiquem diferentes categorias de requisitos, estes podem ser agrupados para que diferentes equipes possam realizar a atividade de derivação de forma paralela (relacionado ao conceito de paralelismo).*
- A03. Análise de viabilidade**, incluindo o levantamento de custos (tempo, financeiro etc.), recursos disponíveis (tecnologias, expertise da equipe etc.) e restrições aplicáveis (legislação), verificando se os recursos são suficientes para se alcançar o objetivo estabelecido e se as restrições não o inviabilizam. **R03.** *Os conceitos de coleta e organização de dados podem ser associados a esta atividade, quando o problema envolver uma gama ampla e diversa de custos e recursos. Ademais, estes dados devem ser analisados para identificar a viabilidade de se resolver o problema (avaliação de dados). Essa avaliação pode ser realizada sob diferentes aspectos (tempo, custo etc.), o que torna necessária a abstração das informações não relevantes àquele aspecto que está sendo analisado. Essas análises podem ser feitas simultaneamente, remetendo também ao conceito de paralelismo.*
- A04. Validação dos requisitos** identificados, observando *corretude, completude e coerência (mantendo somente os requisitos necessários)*, realizando ajustes quando necessário. *Nesta etapa deve-se assegurar que todos os requisitos e somente os necessários sejam considerados.* **R04.** *A validação dos requisitos se relaciona com o PC por meio da avaliação da correção e verificação de propriedades.*
- A05. Estruturação da resolução do problema**, identificando os subproblemas e seus relacionamentos. *A forma em que esses subproblemas se relacionam deve ser estabelecida com base na dependência que cada subproblema tem dos demais, o que determinará a forma em que as suas soluções serão compostas.* **R05.** *A estruturação da resolução do problema inclui a sua decomposição em subproblemas, diretamente relacionado ao conceito de decomposição do PC. Além disso, podem ser identificados subproblemas que seguem um mesmo padrão para que se possa generalizar a solução que os resolve, relacionado ao conceito de generalização.*
- A06. Planejamento do projeto**, elaborando o cronograma de resolução dos subproblemas, levando em conta as dependências entre eles (*já identificadas*) e registrando os possíveis riscos envolvidos, bem como alocando a equipe de execução para cada etapa discriminada no cronograma. **R06.** *A definição do cronograma envolve a identificação da ordem na qual as ações relacionadas as demais atividades serão realizadas (o que está relacionado com o conceito de pensamento algorítmico, como por exemplo, condicionais) e a alocação da equipe para realizar as*

diferentes ações. Neste planejamento devem ser identificadas ações que podem ser realizadas em paralelo, definindo fluxos independentes de execução, destacando pontos de sincronização (*paralelismo*). *Na identificação dos riscos, ainda se tem a relação com o conceito de avaliação.*

**A07. Descrição abstrata da solução de cada subproblema**, identificando os objetivos, as restrições e as possibilidades de reuso.

**A08. Definição das interfaces dos subproblemas identificados**, determinando os recursos necessários e resultados esperados de forma precisa, *levando em conta os relacionamentos definidos entre os subproblemas, com a finalidade de explicitar as informações que são relevantes aos demais componentes.* **R08.** *Nesta etapa, são identificadas as informações/recursos (entradas e saídas) relevantes associadas a cada subproblema, ignorando as demais. Deste modo a atividade está relacionada com o conceito de abstração.*

**A10. Integração das soluções.** **R10.** *Esta atividade se relaciona com PC por meio do conceito de decomposição (composição). Essa composição deve ser descrita de modo a se obter um (ou mais) algoritmo(s) que constitua(m) a solução detalhada do problema original (pensamento algorítmico). Se for o caso, a resolução do(s) algoritmo(s) descrito(s) pode(m) ser implementada(s) (automação).*

**A11-A12. Verificação da resolução do(s) (sub)problema(s).** **R11-R12.** *A relação com o PC se dá por meio do conceito de avaliação. A avaliação pode se dar através de testes (dos subproblemas, da integração dos mesmos e do problema), os quais consistem em verificar se, seguindo os passos da resolução com recursos dados, é possível chegar no objetivo estabelecido. Na seleção de casos de teste é possível identificar a relação com a abstração, ao escolher as características relevantes que permitem categorizar os recursos de entrada (dados) em classes.*

**A13. Execução da solução do problema em um ambiente real**, envolvendo os atores relacionados. **R13.** *Os passos descritos na resolução do problema são realizados, remetendo ao conceito de execução de etapas do pensamento algorítmico.*

## 5. Conclusão

Este trabalho buscou validar por meio de um painel de especialistas uma proposta de caracterização de conceitos do PC em processos da ES. A pesquisa realizada incluiu avaliações quantitativas e qualitativas. Os resultados apresentados na análise dos dados quantitativos mostram que os especialistas concordaram com a caracterização estabelecida. Já a análise dos dados qualitativos pontuou questões que permitiram adequar e aperfeiçoar as caracterizações, o que aumenta a confiabilidade das relações descritas. Ao considerar especialistas de ambas as áreas, PC e ES, foi possível adequar os termos utilizados para uma descrição mais clara e objetiva. Pelas percepções dos respondentes, pode-se dizer que em todas as atividades da ES consideradas é viável caracterizar diferentes conceitos do PC. Apesar de nenhum dos avaliadores ter mencionado, é possível que existam relações não identificadas. Os resultados deste trabalho permitirão descrever uma proposta que permita sistematizar a elaboração de projetos educacionais, onde os estudantes além de conseguirem se guiar para o desenvolvimento do projeto, independente do tema, conseguirão desenvolver habilidades do PC.

Como trabalho futuro, pretende-se fazer uma nova rodada de avaliações, apresentando as reformulações estabelecidas e buscando obter um consenso ainda maior entre os especialistas.

## Referências

- Aidas, V., Vasilienė-Vasiliauskienė, V., Meidute-Kavaliauskiene, I., and Sabaitytė, J. (2020). Peculiarities of educational challenges implementing project-based learning. *World Journal on Educational Technology: Current Issues*, 12:136–149.
- Apiola, M. and Sutinen, E. (2020). Design science research for learning software engineering and computational thinking: Four cases. *Computer Applications in Engineering Education*.
- Brennan, K. and Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada*, volume 1, page 25.
- Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., and Wollard, J. (2015). Computational thinking - A guide for teachers. Guide, Computing at School. [https://eprints.soton.ac.uk/424545/1/150818\\_Computational\\_Thinking\\_1\\_.pdf](https://eprints.soton.ac.uk/424545/1/150818_Computational_Thinking_1_.pdf).
- CSTA/ISTEA (2011). Computational Thinking in K–12 Education: leadership toolkit. [https://cdn.iste.org/www-root/2020-10/ISTE\\_CT\\_Leadership\\_Toolkit\\_booklet.pdf](https://cdn.iste.org/www-root/2020-10/ISTE_CT_Leadership_Toolkit_booklet.pdf).
- Dalkey, N. and Helmer, O. (1963). An experimental application of the delphi method to the use of experts. *Management science*, 9(3):458–467.
- Dyah Utami, D., Setyosari, P., Fajarianto, O., Kamdi, W., and Ulfa, S. (2023). The correlation between metacognitive and problem solving skills among science students. *EduLine: Journal of Education and Learning Innovation*, 3:138–143.
- Fronza, I., El Ioini, N., and Corral, L. (2016). Teaching software design engineering across the k-12 curriculum: Using visual thinking and computational thinking. In *Proceedings of the 17th Annual Conference on Information Technology Education, SIGITE '16*, page 97–101, New York, NY, USA. Association for Computing Machinery.
- Fronza, I., Ioini, N. E., and Corral, L. (2017). Teaching computational thinking using agile software engineering methods: A framework for middle schools. *ACM Trans. Comput. Educ.*, 17(4).
- Ignatyeva, G. A., VILKOVA, A. V., and TIMOFEEVA, E. (2018). Educational project as a way of improving students creative activity. *Revista Espacios*, 39(25).
- Kayo, E. K. and Securato, J. R. (1997). Método delphi: fundamentos, críticas e vieses. *Caderno de Pesquisas em Administração*, 1(4):51–61.
- Linstone, H. A. and Turoff, M. (2002). *The Delphi method: techniques and application*. Addison Wesley.
- Nielsen, J. and Landauer, T. K. (1993). A Mathematical Model of the Finding of Usability Problems. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems, CHI '93*, page 206–213, New York, NY, USA. Association for Computing Machinery.
- OECD (2018). The future of education and skills: Education 2030. the future we want. OECD Publishing.

- Ribeiro, L., Foss, L., and da Costa Cavalheiro, S. A. (2020). Entendendo o pensamento computacional. In Raabe, A., Zorzo, A. F., and Blikstein, P., editors, *Computação na educação básica: fundamentos e experiências*. Penso Editora.
- Saad, A. (2022). The effectiveness of project based learning with computational thinking techniques in a software engineering project course. *Journal of Contemporary Issues and Thought*, 12(1):65–79.
- Santos, J., Cavalheiro, S., Foss, L., and Jr., L. R. (2022). Pensamento computacional e engenharia de software: primeiros passos em direção a uma proposta de sistematização de resolução de problemas. In *Anais do XXX Workshop sobre Educação em Computação*, pages 451–462, Porto Alegre, RS, Brasil. SBC.
- Shute, V. J., Sun, C., and Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, 22:142–158.