

Currículo por Competência: Um Relato de Experiência na Reestruturação do Projeto Pedagógico de um Curso de Bacharelado em Engenharia de Software

Gustavo J. de Souza Santos¹, Alinne C. Corrêa Souza¹, André R. Ortoncelli¹, Franciele Beal¹, Rafael A. P. de Oliveira¹, Tatiane Peratz¹, Rodolfo A. Silva¹

¹Coordenação do Curso de Engenharia de Software
Universidade Tecnológica Federal do Paraná – Dois Vizinhos – PR – Brasil

{gustavosantos, alinnesouza, ortoncelli, fbeal, raoliveira, tatianeperatz, rodolfoa}@utfpr.edu.br

Abstract. *This article reports on the experience of the Structuring Teaching Nucleus of a Bachelor's Degree in Software Engineering Course at a Higher Education Institution, in the process of building a new competency-based curriculum matrix. For the development of competencies, the National Curriculum Guidelines and institutional legislation were considered. As contributions of the study, it is highlighted that the competency-based approach allowed: (i) the creation of a Curricular Units matrix aligned with the labor market; and (ii) the natural incorporation of feedback related to students' absorption of knowledge and skills. Thus, this study may be used as a guide for other courses to adopt similar approaches when reformulating their curriculum matrices.*

Resumo. *Este artigo relata a experiência do Núcleo Docente Estruturante de um curso de Bacharelado em Engenharia de Software de uma Instituição de Ensino Superior, no processo de construção da nova matriz curricular em uma abordagem por competências. Para o desenvolvimento das competências, foram consideradas as Diretrizes Curriculares Nacionais e atos normativos institucionais. Como contribuições do estudo, destaca-se que a abordagem baseada em competências permitiu: (i) a criação de uma matriz de Unidades Curriculares alinhada com mercado de trabalho; e (ii) a incorporação natural de feedbacks relacionados à absorção de conhecimento e competências por parte dos estudantes. Dessa forma, este estudo pode ser utilizado como guia para que outros cursos adotem abordagens similares ao reformularem suas matrizes curriculares.*

1. Introdução

O currículo de um curso de graduação reflete todas as experiências de aprendizagem planejadas e pautadas pelas ideias de uma instituição, para um determinado período de tempo, indicando o quê e como os alunos devem aprender, devendo ser revisado constantemente para que se mantenha atualizado com as demandas da sociedade [Buitrago-Florez et al. 2022].

Entre os fatores que motivam a atualização do Projeto Pedagógico de Curso (PPC), destacam-se: (i) adequação à evolução da indústria e das tecnologias [Ellahi et al. 2019]; (ii) garantia da qualidade da instituição de ensino [Brooman et al. 2015]; (iii) melhoria do processo de ensino e aprendizagem [Twigg 2013]; e (iv) adequação à diretrizes curriculares impostas pela instituição ou por diretrizes nacionais. Dessa forma, reformular o PPC

é uma tarefa complexa, sendo aconselhável o uso de um processo sistemático adequado que possa estar alinhada com fatores diversos, por exemplo, características regionais da instituição de ensino e legislação institucional. Para tal, deve-se contar com envolvimento dos docentes e também atentar-se com a complexidade dos processos de construção, para evitar resultados insatisfatórios [Voogt et al. 2019].

Relatos recentes sobre a reformulação de currículos para graduação evidenciam a importância de estratégias focadas em competências e objetivos de aprendizagem. O desenvolvimento de competências é tratado sob diferentes perspectivas, tais como, formação de currículos, avaliação do processo de ensinar e aprender, avaliação de competências e competências docentes [Nunes et al. 2012]. A avaliação da aprendizagem numa abordagem por competências, definida por Scallon [Scallon 2017], tem fundamentado a elaboração de matrizes curriculares de diferentes cursos [Souza 2019, Mognon 2020].

O presente artigo apresenta o relato de experiência do processo de construção da nova matriz curricular do curso de Bacharelado em Engenharia de Software (BES) da Universidade Tecnológica Federal do Paraná (UTFPR), Campus Dois Vizinhos. No artigo serão apresentadas as atividades realizadas pelo Núcleo Docente Estruturante (NDE) do BES e os resultados obtidos na construção dessa matriz em uma abordagem por competências. O Projeto Pedagógico de Curso (PPC) foi aprovado em 27 de dezembro de 2022 pela Resolução nº 224 do Conselho de Graduação da UTFPR e começou a ser implementado no primeiro semestre de 2023 [UTFPR 2022a].

O restante deste trabalho está organizado da seguinte forma. A Seção 2 apresenta a fundamentação teórica para o processo de reformulação do PPC. O processo para a criação da matriz é apresentado na Seção 3. A Seção 4 descreve os resultados alcançados pela reformulação, bem como as lições aprendidas desse processo. Por fim, na Seção 5 são apresentadas as considerações finais e recomendações apresentadas pelos autores para futuros estudos referentes ao processo de elaboração da matriz por competência.

2. Fundamentação Teórica

2.1. Taxonomia de Bloom

A Taxonomia de Bloom consiste em um conjunto de categorias hierárquicas que descreve diferentes tipos de aprendizagem cognitiva, muito utilizada em diferentes campos da educação. Foi criada em 1956 por Benjamin Bloom e seus colaboradores e é uma ferramenta que oferece base para desenvolvimento de instrumentos de avaliação, assim como a criação de mecanismos de integração de habilidades simples para domínio de habilidades mais complexas [Momen et al. 2022].

Esta taxonomia, em sua versão revisada [Anderson and Krathwohl 2001], é composta por seis níveis de aprendizagem cognitiva organizados em ordem crescente de complexidade. Para atingir um objetivo cognitivo em um nível mais alto, subentende-se que os objetivos de níveis anteriores foram atingidos e mobilizados. Estes níveis vão de habilidades mais simples como *Lembrar* ou recordar informações previamente aprendidas, passando por *Compreender*, *Aplicar*, *Analisar*, *Avaliar* e, por fim, *Criar* ou projetar novos produtos. A construção da matriz curricular do curso de BES utilizou a Taxonomia de Bloom revisada (ver Seção 4) na criação dos objetivos de aprendizagem das unidades curriculares. Esses objetivos auxiliam os docentes no planejamento e desenvolvimento de atividades que exercitem os objetivos de compreensão e desenvolvimento cognitivo.

2.2. Design de Curso por Competência

O design por competência é uma abordagem que se concentra no desenvolvimento de habilidades e na mobilização de conhecimentos. Neste trabalho, utiliza-se a definição de competência adotada por Scallon [Scallon 2017], que refere à “possibilidade, para um indivíduo, de mobilizar de maneira interiorizada um conjunto integrado de recursos em vista de resolver uma família de situações-problema” [Roegiers and Ketele 2001]. Ou seja, os estudantes são preparados para atuar em situações variadas que se aproximam do cotidiano da profissão ou ocupação aprendida, por meio de metodologias de ensino e de avaliação que permitam a aquisição dessas competências.

Nessa abordagem, distinguem-se três tipos de conhecimentos: i) os declarativos (ou *saberes*) que correspondem a saberes teóricos e formais, como regras, técnicas, normas, etc.; ii) os procedimentais (ou *saber-fazer*) que envolvem a combinação adequada dos saberes e que se desenvolvem na prática da profissão ou ocupação aprendida; e iii) os condicionais (ou *saber-ser*) que inclui aspectos afetivos, sociais, de visão de mundo, tais como autonomia, senso crítico, criatividade, colaboração, entre outros.

No conjunto dos trabalhos de Anderson e Krathwohl [Anderson and Krathwohl 2001], a utilização de uma organização curricular por competências pressupõe uma mudança de paradigma no Ensino da Engenharia no Brasil, instituída particularmente por meio das habilidades esperadas do egresso nas Diretrizes Curriculares Nacionais (DCN's) para cursos da Computação, que inclui o curso de Engenharia de Software [MEC 2016]. Nesse sentido, a construção de competências ao longo da matriz curricular do curso de BES baseou-se no desenvolvimento e integração entre saber, saber-fazer e saber-ser, comunicando de forma clara o que é esperado do estudante a cada etapa de sua jornada de aprendizagem.

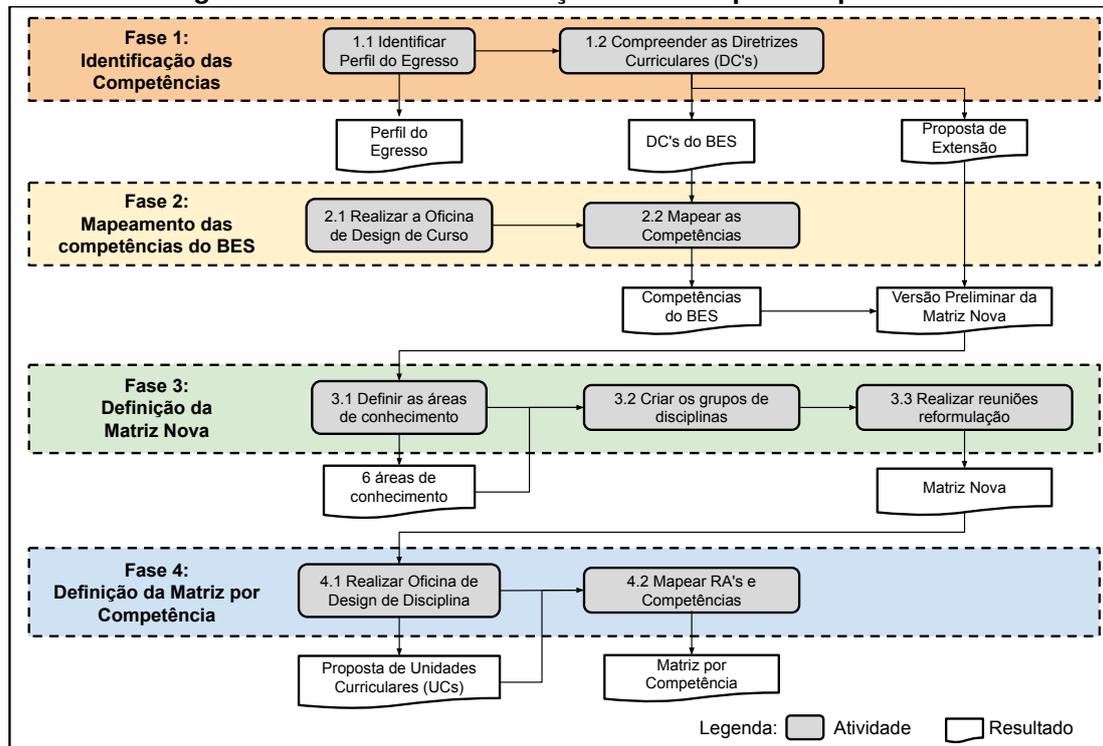
3. Metodologia

A organização didático-pedagógica de um curso superior deve considerar legislações e regulamentações definidas em âmbito nacional e institucional. A construção do PPC do BES foi fundamentada, principalmente no Plano de Desenvolvimento Institucional (PDI) e Projeto Político Institucional (PPI) e Diretrizes Curriculares (DCs) para os Cursos de Graduação da UTFPR [UTFPR 2017, UTFPR 2019, UTFPR 2022b]. Em âmbito nacional, foram consideradas legislações que regulamentam os cursos de Bacharelado em instituições federais de ensino superior, principalmente as DCNs [MEC 2012, MEC 2016].

A reestruturação do PPC consistiu na elaboração de uma matriz curricular que atendesse ao perfil do egresso do curso, estruturada a partir de um currículo por competências. O processo de construção da matriz por competência (Figura 1) foi composto por quatro fases e nove atividades descritas a seguir:

- 1.1 **Identificar o Perfil do Egresso:** o perfil do egresso foi identificado a partir das DCNs características, habilidades e áreas de atuação no mercado.
- 1.2 **Compreender as Diretrizes Curriculares:** a partir das diretrizes supracitadas, buscou-se, junto ao NDE, as alterações necessárias na matriz curricular de modo a atendê-las. Tais alterações incluíram a curricularização de atividades de extensão, a possibilidade e operação de Unidades Curriculares (UCs) à distância, a execução de

Figura 1. Processo de construção da matriz por competência.



componentes curriculares de estágio e atividades complementares, o cumprimento de carga horária de humanidades e a formação voltada para o desenvolvimento de competências na área de Computação.

- 2.1 **Realizar Oficina de Design de Curso:** durante os meses de Setembro e Outubro de 2019, representantes dos NDEs de cursos da área de Computação da UTFPR reuniram-se em oficinas para reformulação dos PPCs. A oficina seguiu a definição de competências de Scallon [Scallon 2017] e a taxonomia de Bloom revisada [Anderson and Krathwohl 2001]. A partir desses encontros, foram elaboradas as competências comuns entre cursos da Computação, as competências específicas de cada curso e elementos de competência derivados de ambas.
- 2.2 **Mapeamento de Competências:** as competências elaboradas nas oficinas foram validadas pelo NDE e mapeadas com as habilidades instituídas nas DCNs do MEC para os cursos de Computação [MEC 2012, MEC 2016]. Buscou-se estabelecer o cumprimento de todas as habilidades das DCNs, assim como uma proposta inicial de matriz curricular para os primeiros semestres do curso, visto que nestes semestres são empenhadas as competências básicas, como a resolução de problemas estruturados e implementação de soluções para esses problemas, entre outras.
- 3.1 **Definir as áreas de conhecimento:** de modo a facilitar a reformulação da matriz curricular, foram criadas áreas de conhecimento de acordo com o corpo de conhecimento sobre Engenharia de Software, ou SWEBOK [Society et al. 2014].
- 3.2 **Criar Grupos de Disciplinas:** a partir das áreas de conhecimento definidas foram criados os grupo de disciplinas. Cada grupo foi responsável por elaborar propostas de organização para as UCs da matriz, incluindo carga horária teórica e prática, temas de estudo, pré-requisitos, e relação com outras UCs do curso.

- 3.3 **Realizar Reuniões de Reformulação da Matriz:** quinzenalmente o NDE reuniu-se para avaliar propostas de alteração da matriz curricular, adequação às DCs, propostas para extensão, entre outras. As propostas dos grupos de disciplinas eram discutidas regularmente nessas reuniões, levando em consideração que o NDE possui autonomia para adotar as alterações propostas, com a aprovação da maioria dos integrantes e avaliações registradas em ata. Esta atividade foi a mais dispendiosa do processo de construção, uma vez que alterações na matriz impactavam a organização das demais UCs e geralmente implicavam em aumento de carga horária semanal ou semestral. Além disso, devido a restrições impostas pela pandemia COVID-19, a maioria das reuniões foram conduzidas de forma remota, o que onerou o debate e avaliação de propostas de alteração.
- 4.1 **Realizar Oficina de Design de Disciplina:** a proposta de matriz curricular nova foi apresentada aos docentes, destacando as motivações para reformulação e as alterações aplicadas na matriz. Em seguida, foi organizada junto ao Núcleo de Ensino da UTFPR, uma oficina aos docentes, com foco na estruturação das UCs em temas de estudo e Resultados de Aprendizagem (RA) [Biggs and Tang 2011]. O RA esclarece de forma objetiva e mensurável o que se espera do estudante ao final de um processo de ensino, ou seja, de uma UC. Após a formação, cada docente foi responsável por reformular as ementas das UCs que ministrava, com apoio do grupo de disciplina correspondente àquela UC.
- 4.2 **Mapear os RAs e Competências:** a elaboração dos RAs auxiliou a última etapa da construção do currículo, que consistiu no mapeamento entre (i) o que é esperado do estudante em uma UC, e (ii) a competência esperada do egresso, permitindo assim que o docente estabeleça o quanto um estudante está próximo de atingir uma determinada competência. Este mapeamento também é importante para que o corpo docente, e especificamente o NDE, constate como cada competência está sendo trabalhada ao longo do curso, mobilizando conhecimentos e habilidades para a resolução de problemas autênticos ao ramo de atuação do egresso.

4. Resultados

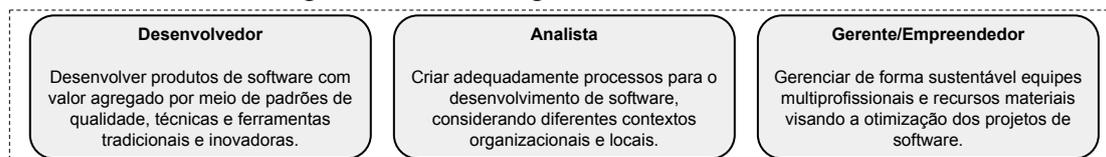
Com a aplicação do processo de construção da matriz curricular apresentado na seção anterior, é possível elencar os resultados obtidos em cada fase do processo.

4.1. Fase 1: Identificação das Competências

Nesta fase, os principais resultados alcançados estão relacionados com a identificação do(a): (i) Perfil do Egresso; (ii) Adequação da Matriz Curricular para atender às legislações; e (iii) Proposta Inicial para Extensão. Um dos componentes norteadores do PPC é o **Perfil do Egresso**, que estabelece as habilidades e competências que o egresso deve adquirir ao longo do curso para sua inserção no mercado de trabalho. Neste contexto, o egresso do curso de BES deve ser “um profissional capaz de criar processos, serviços e/ou produtos considerando o ciclo de desenvolvimento de software”. Esse profissional pode atuar como gerente, analista, arquiteto, programador e testador de soluções de software nas organizações, conforme é apresentado na Figura 2.

A identificação das **Diretrizes Curriculares** para o BES da UTFPR foram fundamentais para o processo de reformulação, uma vez que o curso busca o desenvolvimento de competências profissionais por meio de situações e atividades autênticas ao

Figura 2. Perfil do Egresso do Curso de BES.



ramo de atuação do egresso. Portanto, para o desenvolvimento de competências profissionais dentro do BES, foram utilizadas as DCNs para cursos de Engenharia de Software [MEC 2016], as DCs para os cursos de graduação da UTFPR [UTFPR 2022b], PDI [UTFPR 2017], PPI [UTFPR 2019] e demais resoluções e portarias da UTFPR, conforme é ilustrado na Tabela 1.

Tabela 1. Adequações do PPC do BES de acordo com as legislações internas e externas.

	Resolução ou Portaria	Requisitos Contemplados no PPC
DCNs	<i>Resolução CNE/CES nº 5/2016</i>	- Carga horária mínima de 3200 horas, distribuída em UCs, Estágio Curricular Obrigatório, Trabalho de Conclusão de Curso e Atividades Complementares. - 40 horas de atividades complementares. - Diretrizes para matrizes de curso de Engenharia de Software
DC's UTFPR	<i>Resolução COGEP nº 167/2022</i> <i>Resolução COGEP nº 142/2022</i>	- Mínimo de 10% do total da carga horária curricular em atividades de extensão (330 horas). - 10% da carga horária de UC's em Humanidades (270 horas). - Carga horária mínima de 360 horas para estágio curricular dos cursos de bacharelado.
UTFPR	<i>Portarias nº 477/2017, nº 766/2017 e nº 971/2017 (e suas alterações)</i> <i>Portarias do reitor nº 1771/2017 e nº 2362/2017</i> <i>Resolução conjunta nº 1/2020.</i>	PDI PPI Regulamento dos Estágio Curriculares Supervisionados dos Cursos de Bacharelado.

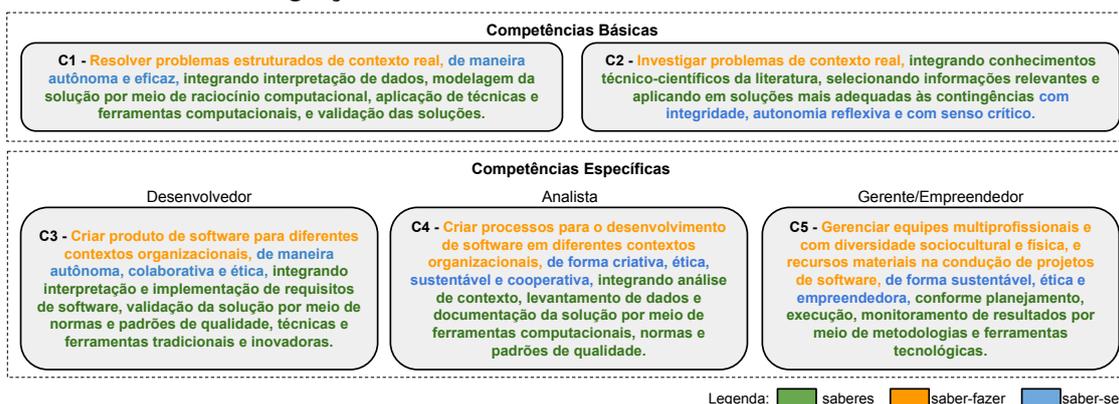
Ao final desta fase, foi elaborada a **Proposta para Extensão**. O PPC prevê o cumprimento da extensão por meio das UCs de Fábrica de Software 1, Fábrica de Software 2 e Design de Software, totalizando 330 horas de carga horária. Nas Fábricas de Software, os estudantes são apresentados a problemas reais de empresas ou comunidade externa, em uma proposta que está alinhada com ações inovadoras no ensino da Engenharia de Software [dos Santos et al. 2021]. Em Design de Software, os estudantes devem planejar soluções sem a necessidade da execução do projeto; dessa forma, são permitidas soluções que necessitam de recursos de hardware ou software ainda não disponíveis, de modo a fomentar projetos que possam ser executados nas UCs de Fábrica de Software.

4.2. Fase 2: Mapeamento das Competências do BES

Nesta fase, os resultados alcançados estão diretamente relacionados à definição das **Competências do BES**. A organização do perfil do egresso por meio de competências permite que a certificação dessas competências seja realizada de forma gradual durante a integralização do discente no curso. Essa certificação é realizada por meio de *UCs integradoras*, previstas na matriz curricular e orientadas à resolução de problemas reais a partir de demandas reais da comunidade, a exemplo das UC's propostas para extensão na fase anterior. Dessa forma, é previsto que, nestas UCs, o discente certifique que atingiu determinada competência.

Um total de cinco competências foram criadas, sendo divididas em duas competências básicas (C_1 , C_2), *i.e.*, comuns a todos os cursos de Computação da UTFPR, e três competências específicas (C_3 , C_4 e C_5), *i.e.*, referentes a cada um dos perfis do egresso do BES, conforme apresentado na Figura 3.

Figura 3. Competências Básicas e Específicas do Curso de BES agrupadas de acordo com a integração de diferentes saberes.



A partir das competências listadas, foram identificados os seus respectivos elementos, bem como o período certificador. A competência C_1 é certificada no quarto período e possui cinco Elementos de Competência (ECs); C_2 é certificada no quinto período e possui oito ECs; C_3 é certificada no sexto período e possui quatro ECs; C_4 é certificada no sexto período e possui sete ECs; e C_5 é certificada no sétimo período e possui cinco ECs. A competência C_3 relacionada ao perfil de “Desenvolvedor”, por exemplo, possui os seguintes ECs:

- **EC 3.1:** “Interpretar requisitos de software para diferentes contextos organizacionais buscando referências atualizadas na área”;
- **EC 3.2:** “Empregar técnicas e ferramentas tecnológicas adequadas para a implementação dos requisitos de software especificados”;
- **EC 3.3:** “Implementar produtos de software para diferentes contextos organizacionais de maneira autônoma e colaborativa”; e
- **EC 3.4:** “Validar a solução de acordo com os requisitos especificados, e normas e padrões de qualidade com ética”.

4.3. Fase 3: Definição da Nova Matriz

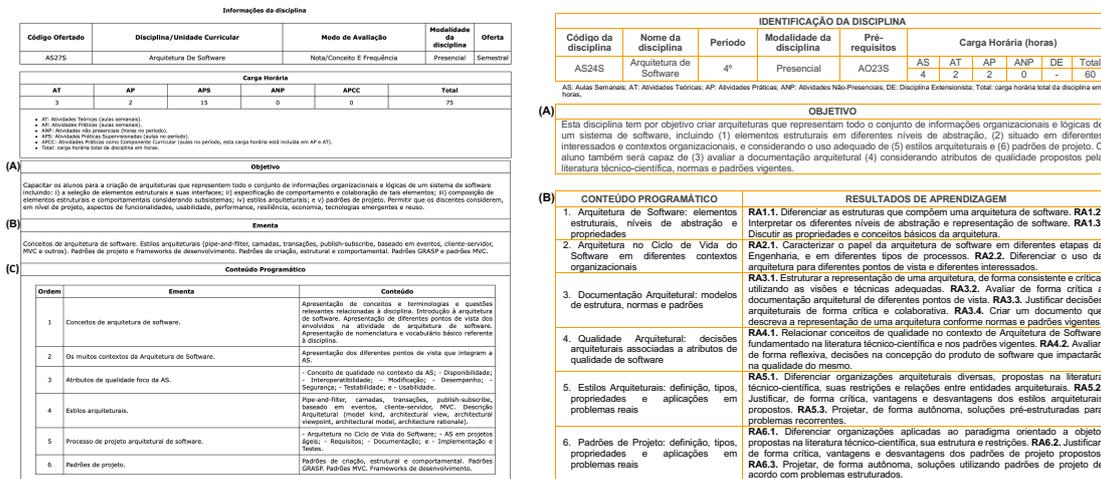
Para a construção da matriz, um ponto inicial foi a definição das **áreas de conhecimento** para criação de grupos de disciplinas. Neste contexto, visando atender o estabelecido na resolução CNE/CES nº 5, as UCs do BES estão organizadas em torno de seis áreas de conhecimento, estruturadas a partir do SWEBOK [Society et al. 2014]: (i) Fundamentos Matemáticos aplicados à Computação, (ii) Fundamentos de Análise e Programação, (iii) Aspectos Humanos aplicados à Engenharia de Software, (iv) Fundamentos de Ciência da Computação, (v) Métodos e Técnicas aplicados à Engenharia de Software, e (vi) Síntese e Integração do Conhecimento.

Com a áreas de conhecimento definidas, seis grupos de disciplinas foram criados para facilitar a reestruturação das UCs na matriz. A nova matriz, organizada por área de conhecimento pode ser acessada por meio do link encurtador.com.br/zFGU8.

4.4. Fase 4: Definição da Matriz por Competência

A reformulação do curso do BES por competência estimulou o NDE a elaborar um novo Plano de Ensino com a finalidade de atender os conhecimentos declarativos, procedimentais e condicionais (ou saber, saber-fazer e saber-ser). Enquanto o plano anterior focava na questão “Qual conteúdo cada estudante deveria saber ao final dessa disciplina?”, o novo plano atende à questão “O que cada estudante deveria saber fazer?”.

Para implantar essa nova proposta, foi necessário o apoio e a participação do corpo docente que ministra aulas no curso. Para isso, foi realizada uma Oficina de Design de Disciplinas para instruir cada docente na **reformulação das UCs** em temas de estudo e RAs, utilizando a Taxonomia de Bloom revisada. Um exemplo da proposta da UC “Arquitetura de Software” antes e após a reformulação usando temas de estudo e resultados de aprendizagem é apresentado na Figura 4.



(a) Unidade curricular antes da reformulação. (b) Reformulação utilizando temas de estudo e RAs.

Figura 4. Exemplo da UC “Arquitetura de Software” antes (a) e após (b) a reformulação da matriz, por meio da formulação de resultados de aprendizagem.

Na Figura 4 é possível notar que o item A da Figura 4(b) foi construído usando temas de estudo contemplando os itens A e B da Figura 4(a). Cada tema de estudo foi transformado em um conteúdo programático e, para cada conteúdo, foram identificados os seus respectivos RAs, conforme apresentado no item B da Figura 4(b). Assim, o item C da Figura 4(a) foi substituído pelo item B da Figura 4(b).

O NDE tem conhecimento que os métodos ativos de aprendizagem e os métodos avaliativos são elementos essenciais na construção da formação por competências. No que tange aos métodos ativos de aprendizagem, o NDE não definiu quais deveriam ser seguidos em cada UC, pois eles já são aplicados pelos docentes do BES, e.g., Aprendizagem Baseada em Problemas, Aprendizagem Baseada em Projetos, bem como projetos interdisciplinares e integradores que são desenvolvidos nas Fábricas de Software. Quanto aos métodos avaliativos, os docentes têm a liberdade para definir a melhor forma de medir o aproveitamento dos seus discentes, mediante provas, exercícios, trabalhos, projetos ou outras atividades condizentes com a UC ministrada e os RAs esperados.

Com base nas UCs reformuladas, o resultado final do processo apresentado na Fi-

gura 1 foi a geração da **matriz por competência**. Todos os RAs propostos para as UCs foram devidamente alinhados aos ECs a serem desenvolvidas ao longo do curso. Essa relação possibilita que o NDE acompanhe como as competências são desenvolvidas em cada UC de modo acumulativo e contínuo no transcorrer do curso. A Figura 5 apresenta um exemplo de uma seleção dos RAs da UC “Arquitetura de Software” mapeados de acordo com os ECs da competência básica C_2 e a competência específica C_3 . O mapeamento das competências e RAs de todas as UCs do BES pode ser acessado por meio do link encurtador.com.br/aioHU.

COMPETÊNCIA BÁSICA 2	C2 - Investigar problemas de contexto real integrando conhecimentos técnico-científicos da literatura, selecionando informações relevantes e aplicando em soluções mais adequadas às contingências com integridade, autonomia reflexiva e com senso crítico. Certificação: 5º período							
ELEMENTOS DE COMPETÊNCIA	Descrever situações envolvendo problemas de contexto real e fundamentado na literatura técnico-científica e em fontes fidedignas.	Problematizar situações de contexto real descritas	Formular hipóteses a partir de problemas selecionados	Definir métodos e estratégias ou procedimentos adequados para testar a hipótese	Validar hipóteses aplicando métodos, estratégias ou procedimentos definidos	Analisar resultados de validação, replanejando método, estratégia ou procedimento de forma reflexiva	Argumentar, de forma consistente e crítica e pautado em conhecimentos técnico-científicos da literatura, em defesa de resultados analisados.	Documentar o processo de desenvolvimento de projetos
UC - ARQUITETURA DE SOFTWARE	RA3.1. Estruturar a representação de uma arquitetura, de forma consistente e crítica, utilizando as visões e técnicas adequadas.		RA3.3. Justificar decisões arquiteturais de forma crítica e colaborativa.	RA5.3. Projetar, de forma autônoma, soluções pré-estruturadas para problemas recorrentes. RA6.3. Projetar, de forma autônoma, soluções utilizando padrões de projeto de acordo com problemas estruturados.	RA3.2. Avaliar de forma crítica a documentação arquitetural de diferentes pontos de vista.		RA5.2. Justificar, de forma crítica, vantagens e desvantagens dos estilos arquiteturais propostos. RA6.2. Justificar, de forma crítica, vantagens e desvantagens dos padrões de projeto propostos.	

(a) RAs mapeados de acordo com a competência básica C_2 .

COMPETÊNCIA ESPECÍFICA: DESENVOLVEDOR	C3 - Criar produto de software para diferentes contextos organizacionais, de maneira autônoma, colaborativa e ética, integrando interpretação e implementação de requisitos de software e validação da solução por meio de normas e padrões de qualidade, técnicas e ferramentas tradicionais e inovadoras. Certificação: 5º período			
ELEMENTOS DE COMPETÊNCIA	Interpretar requisitos de software para diferentes contextos organizacionais buscando referências atualizadas na área	Empregar técnicas e ferramentas tecnológicas adequadas para a implementação dos requisitos de software especificados	Implementar produtos de software para diferentes contextos organizacionais de maneira autônoma e colaborativa	Validar a solução de acordo com os requisitos especificados, e normas e padrões de qualidade com ética
UC - ARQUITETURA DE SOFTWARE		RA1.1. Diferenciar as estruturas que compõem uma arquitetura de software. RA1.2. Interpretar os diferentes níveis de abstração e representação de software. RA1.3. Discutir as propriedades e conceitos básicos da arquitetura.	RA2.1. Caracterizar o papel da arquitetura de software em diferentes etapas da Engenharia, e em diferentes tipos de processos.	

(b) RAs mapeados de acordo com a competência específica C_3 .

Figura 5. Exemplo de Mapeamento dos RAs da unidade curricular “Arquitetura de Software” do BES em relação à segunda competência básica (C_2) e à terceira competência específica (C_3 - Desenvolvedor) do egresso.

Nas Figuras 5(a) e 5(b), o RA da UC “Arquitetura de Software” destacado em amarelo indica que o EC correspondente é *internalizado* nessa UC; ou seja, o estudante tem o primeiro contato com terminologias e teoria em relação a este EC. Os RAs destacados em verde indicam que o EC correspondente é *mobilizado* nesta UC; ou seja, o estudante terá a oportunidade de utilizar habilidades, técnicas e ferramentas aprendidas, e mobilizá-las em novas situações-problema para atender este RA. Por fim, ECs são *integralizados* em UCs que certificam competências. Como apresentado na Seção 4.2, a competência básica C_2 é certificada no quinto período, por meio da UC “Fábrica de Software 1”, e a competência específica C_3 é certificada no sexto período por meio da UC “Fábrica de Software 2”. Considerando que o itinerário formativo das competências é algo presente no curso, os ECs e RAs devem ser progressivos e refletir a evolução do discente ao longo do curso.

4.5. Lições Aprendidas

A construção de um currículo baseado em competências é uma abordagem educacional cujo enfoque dá-se em torno do processo de ensino-aprendizagem de habilidades e conhecimentos específicos. Tal abordagem é comumente contrastada com modelos de currículos tradicionais que focam em conhecimentos (ou conteúdos) mais generalistas e

de escopo aberto. Com a aplicação dessa abordagem para a construção da nova matriz curricular do curso de Bacharelado em Engenharia de Software da UTFPR *campus* Dois Vizinhos, foi possível levantar vantagens e desvantagens a partir da aplicação de cada fase do processo apresentado neste artigo.

Foram percebidas como **vantagens**: *(i) os objetivos de aprendizagem são claros, facilitando para os estudantes o entendimento das habilidades que devem ser compreendidas, praticadas e aprimoradas; (ii) as habilidades adquiridas têm uma maior relevância para o mercado de trabalho, uma vez que o processo empregado está em total alinhamento com demandas do mundo real; e (iii) o currículo baseado em competência naturalmente incorpora avaliações e feedbacks constantes, permitindo que estudantes monitorem seus progressos, identificando áreas nas quais eles precisam melhorar.*

Foram percebidas como **desvantagens** da abordagem: *(i) o currículo pode ter escopo limitado ao perfil do egresso, deixando alguns estudantes com uma lacuna de conhecimentos, habilidades e expertises em outras áreas de atuação; (ii) o processo avaliativo de algumas disciplinas podem ser atípicas e baseadas em resultados, o que pode levar estudantes a atravessarem processos estressantes de entregas e metodologias novas; (iii) existe um alto custo em preparações, planejamentos e avaliações de disciplinas oriundas de currículos baseados em competências; e (iv) existem alguns limites de flexibilidade que precisam ser administrados de modo a permitir que atualizações na matriz sejam incorporadas ao longo do tempo, como a relação entre UCs e competências do curso.*

5. Considerações Finais

A criação de uma matriz curricular baseada em competências é uma abordagem inovadora de ensino que busca garantir que os estudantes desenvolvam habilidades e conhecimentos essenciais para sua vida acadêmica e profissional. Para auxiliar no processo de criação da matriz, é aconselhável o uso de um processo sistemático adequado para guiar a estruturação da nova matriz. Com isso, os artefatos criados em uma fase são utilizados para o embasamento das atividades nas fases seguintes do processo.

Neste artigo, apresentou-se o processo de construção da matriz por competências utilizado para a criação da nova matriz curricular do curso de Bacharelado em Engenharia de Software da Universidade Tecnológica Federal do Paraná, *campus* Dois Vizinhos. O processo de construção da matriz por competência foi composto por quatro fases e nove atividades, envolvendo o Núcleo Docente Estruturante e todos os docentes do curso. A proposição do currículo baseado em competências apresentou desafios, comparado ao currículo atual, como: *(i) necessidade de mudança de paradigma na maneira como os docentes ensinam e os estudantes aprendem; (ii) inserção de disciplinas extensionistas; (iii) reformulação das UCs usando RAs.* No entanto, observou-se um grande engajamento dos docentes e estudantes envolvidos no processo. Além do compartilhamento de conhecimento por meio dos processos e métodos utilizados, este estudo traz relatos das lições aprendidas durante mais de três anos de reuniões de grupos de trabalhos.

Conclui-se que o currículo baseado em competências é uma abordagem promissora para a educação, pois permite uma aprendizagem mais significativa e relevante para os estudantes, preparando-os melhor para os desafios da vida e do mercado de trabalho. Recomenda-se, assim, que outras instituições educacionais adotem esse modelo para melhorar a qualidade de ensino e aprendizagem dos seus cursos.

Referências

- Anderson, L. W. and Krathwohl, D. R., editors (2001). *A Taxonomy for Learning, Teaching, and Assessing. A Revision of Bloom's Taxonomy of Educational Objectives*. Allyn & Bacon, New York, 2 edition.
- Biggs, J. and Tang, C. (2011). *Teaching for quality learning at university*. McGraw-Hill Education (UK).
- Brooman, S., Darwent, S., and Pimor, A. (2015). The student voice in higher education curriculum design: is there value in listening? *Innovations in education and teaching international*, 52(6):663–674.
- Buitrago-Florez, F., Sanchez, M., Pérez Romanello, V., Hernandez, C., and Hernández Hoyos, M. (2022). A systematic approach for curriculum redesign of introductory courses in engineering: a programming course case study. *Kybernetes*.
- dos Santos, A. S., Neves, M., Rodrigues, Y., de Oliveira, N. H., Kuhn, D., Santos, G., and Silva, R. A. (2021). Experiência do Projeto “Fábrica de Software” em um Curso de Engenharia de Software. In *Escola Regional de Engenharia de Software*, pages 89–98. SBC.
- Ellahi, R. M., Khan, M. U. A., and Shah, A. (2019). Redesigning curriculum in line with industry 4.0. *Procedia computer science*, 151:699–708.
- MEC (2012). Diretrizes curriculares nacionais para os cursos de graduação em computação – ministério da educação – parecer CNE/CES nº 136/2012, aprovado em 8 de março de 2012. Disponível em: <http://portal.mec.gov.br/component/content/article?id=12991>. Brasília, DF, Brasil. Acesso: agosto de 2022.
- MEC (2016). Diretrizes curriculares nacionais para os cursos de graduação na área da computação, abrangendo os cursos de bacharelado em ciência da computação, em sistemas de informação, em engenharia de computação, em engenharia de software e de licenciatura em computação, e dá outras providências. – ministério da educação – resolução num. 5, de 16 de novembro de 2016. Disponível em: <http://portal.mec.gov.br/docman/novembro-2016-pdf/52101-rces005-16-pdf/file>. Brasília, DF, Brasil. Acesso: agosto de 2022.
- Mognon, F. (2020). O processo de construção da matriz de competências do curso de relações públicas da puc-pr: novos olhares no ensino. *Organicom*, 17(32).
- Momen, A., Ebrahimi, M., and Hassan, A. M. (2022). Importance and implications of theory of bloom's taxonomy in different fields of education. In *International Conference on Emerging Technologies and Intelligent Systems*, pages 515–525. Springer.
- Nunes, S. C., Siqueira, L., et al. (2012). O projeto pedagógico ea orientação para a formação por competências: um estudo em curso superior de uma universidade brasileira. *Revista iberoamericana de educación*.
- Roegiers, X. and Ketele, J.-M. d. (2001). *Une pédagogie de l'intégration: compétences et intégration des acquis dans l'enseignement*. De Boeck.

- Scallon, G. (2017). *Avaliação da aprendizagem numa abordagem por competências*. PUCPress.
- Society, I. C., Bourque, P., and Fairley, R. E. (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK(R)): Version 3.0*. IEEE Computer Society Press, Los Alamitos, CA, USA, 3rd edition.
- Souza, S. T. (2019). Matriz curricular por competência: uma forma de incentivar a aprendizagem e o pensamento crítico. *Revista Brasileira de Ensino de Jornalismo*, 9(25):115–130.
- Twigg, C. A. (2013). Improving learning and reducing costs: Outcomes from changing the equation. *Change: The Magazine of Higher Learning*, 45(4):6–14.
- UTFPR (2017). Plano de desenvolvimento institucional da UTFPR – PDI, 2018 – 2022 – documento elaborado pela comissão designada pelas portarias nº 477, de 16/03/2017, nº 766, de 18/04/2017 e nº 971 (e suas alterações) de 19/05/2017. aprovado em 18/12/2017, deliberação 35/2017 pelo conselho universitário. Disponível em: <https://bit.ly/2JjLJgR>. Curitiba, PR, Brasil. Acesso: agosto de 2022.
- UTFPR (2019). Projeto político pedagógico institucional (PPI) da UTFPR – deliberação COUNI nº 14, de 28 de junho de 2019. Disponível em: <https://cloud.utfpr.edu.br/index.php/s/Z3pqMqWkxbsCbLz>. Curitiba, PR, Brasil. Acesso: agosto de 2022.
- UTFPR (2022a). Alteração do projeto pedagógico do curso de bacharelado em engenharia de software, do ”campus” dois vizinhos.– resolução nº 224/2022-COGEP de 27 de dezembro de 2022. Disponível em: https://sei.utfpr.edu.br/sei/publicacoes/controlador_publicacoes.php?acao=publicacao_visualizar&id_documento=3514626&id_orgao_publicacao=0. Curitiba, PR, Brasil. Acesso: março de 2023.
- UTFPR (2022b). Diretrizes curriculares dos cursos de graduação regulares da universidade tecnológica federal do paran  – resolução nº 142/2022-COGEP de 25 de fevereiro de 2022. Disponível em: https://sei.utfpr.edu.br/sei/publicacoes/controlador_publicacoes.php?acao=publicacao_visualizar&id_documento=2803898&id_orgao_publicacao=0. Curitiba, PR, Brasil. Acesso: agosto de 2022.
- Voogt, J., Pieters, J., and Pareja Roblin, N. (2019). Collaborative curriculum design in teacher teams: Foundations. *Collaborative curriculum design for sustainable innovation and teacher learning*, pages 5–18.