

PROPOSTA COLABORATIVA E INTERDISCIPLINAR DE DESENVOLVIMENTO DE SOFTWARES NO CURSO INTEGRADO DE TÉCNICO EM INFORMÁTICA DO CENEP

Augusto César Oliveira de Almeida¹
Radamila Oliveira do Nascimento²
Anna Raquel da Silva Marinho³
Danylla de Medeiros Souza⁴
Amanda Ohana Albuquerque Gomes⁵

RESUMO

Este trabalho versa sobre as dificuldades de aprender novos conhecimentos no contexto do ensino de disciplinas de programação do curso integrado de Técnico em Informática, onde alunos que ingressam no curso se deparam com conhecimentos e práticas distantes de suas vivências escolares até sua entrada no curso, mesmo percebendo que nenhum conhecimento é isolado e independente de outros. Dessa maneira, alunos tendem a ter dificuldades na abstração dos conteúdos, chegando a casos de alunos no ano final do curso sem compreender o que é um software, mesmo que tenha codificado em projetos e atividades das disciplinas durante os 2 últimos anos escolares. Apresenta-se, então, um relato de experiência locado no Centro Estadual de Educação Profissional Senador Jessé Pinto Freire (Cenep), em Natal/RN, no ano de 2022. Trata-se de uma ação interdisciplinar e colaborativa entre as disciplinas de Lógica de Programação e Desenvolvimento de Sistemas com participação dos alunos de 1º e 3º anos do curso de Técnico em Informática para a produção de softwares. A atuação objetivou diminuir as dúvidas em relação às atividades desenvolvidas por programadores profissionais e ao próprio conceito de “software” para o 1º ano, que se encontra no início dos estudos do Técnico em Informática; integrar teoria e prática estudadas nos conteúdos de engenharia de software e interface gráfica no 3º ano, aplicando-as em simulação prática e controlada; criar softwares em atividade interdisciplinar e colaborativa, integrando conhecimento e alunos do 1º ano e 3º do Ensino Médio Integrado ao Técnico em Informática. Apesar de alguns desafios encontrados ao longo da prática, esses objetivos foram alcançados, demonstrando a relevância de utilizar a interdisciplinaridade no ensino de programação.

Palavras-chave: Desenvolvimento de Software, Interdisciplinaridade, Trabalho Colaborativo, Educação Profissional e Tecnológica.

¹ Especialista em Tecnologias Educacionais e Educação a Distância, Instituto Federal de Educação, Ciência e Tecnologia - RN.

Especialista em Docência para a Educação Profissional e Tecnológica, Instituto Federal de Educação, Ciência e Tecnologia - ES, augustotouya@gmail.com;

² Especialista em Tecnologias Educacionais e Educação a Distância, Instituto Federal de Educação, Ciência e Tecnologia - RN, oliveiraradamila@gmail.com;

³ Especialista em Psicopedagogia Clínica, Hospitalar e Institucional, UNINASSAU, raquelmarinho.linfor@gmail.com;

⁴ Especialista em Mídias na Educação, UERN. Especialista em Docência para Educação Profissional e Tecnológica, IFES, medeirosdanylla@gmail.com;

⁵ Pós-graduanda em Games e Gamificação na Educação, UNINTER. Especialista em Formação Docente para EAD, UNINTER. ohana.albuquerque@gmail.com.

INTRODUÇÃO

Sempre que se enfrenta uma nova fase da vida comumente nos deparamos com desafios e dificuldades que necessitam serem superadas para continuarmos nossa trajetória. Isso não é diferente de quando um estudante inicia seu primeiro ano do ensino médio. O aumento das disciplinas, conteúdos mais complexos, atividades e exercícios mais específicos e outras novidades devem ser superados pelos estudantes.

Nesse cenário, o discente reencontra com disciplinas que lhe são familiares: língua portuguesa, matemática, biologia, física e as demais matérias comuns curriculares. Essa familiaridade ameniza o estranhamento com o novo nível de complexidade na etapa do ensino médio. Contudo, ao inserirmos o contexto da educação profissional e tecnológica, mais especificamente o ensino médio integrado ao curso de Técnico em Informática, em casos raros (e deles, muito provavelmente com alunos egressos da escola privada) teremos essa familiaridade com a informática.

Mesmo com iniciativas com o Proinfo, percebemos que não é cultura da escola pública o trabalho com os conhecimentos vindos da informática. Podemos perceber que o estudante que ingressa na modalidade do técnico integrado dificilmente teve uma experiência prévia com assuntos no âmbito da programação, redes ou manutenção.

Essa falta de experiências com a área da informática pode dificultar as relações mentais que os alunos normalmente fazem com os conteúdos. O estudante saber que aprendendo seu idioma e suas regras pode escrever um livro ou postar textos nas redes sociais, ele percebe que aprendendo língua inglesa ou espanhola poderá compreender e consumir mais conteúdos internacionais e até conseguir um emprego numa multinacional, mas esse mesmo aluno tem dificuldades em relacionar os comandos escritos por ele a um produto final (um *software*).

O presente artigo busca relatar experiências de ensino ocorridas no segundo semestre letivo de 2022 entre as disciplinas de Desenvolvimento de Sistemas e Lógica de Programação, ofertadas em um curso Técnico em Informática integrado ao ensino médio ofertado pelo Centro Estadual de Educação Profissional Senador Jessé Pinto Freire (CENEP).

METODOLOGIA

Este artigo segue a metodologia de relato de experiência com abordagem qualitativa (Severino, 2007). A experiência teve como público alvo alunos do 1º e 3º anos do curso integrado de Técnico em Informática do Centro Estadual de Educação Profissionalizante Senador Jessé Pinto Freire, em Natal/RN, do ano de 2022, cursando respectivamente as

disciplinas de Lógica de Programação e Desenvolvimento de Sistemas. A ação foi aplicada durante a 2º metade do 4º bimestre do mesmo ano.

REFERENCIAL TEÓRICO

Dificuldade no aprendizado de programação

Atualmente, no Brasil, o aprendizado da Informática ou mais especificamente, da programação, está restrito apenas às pessoas que buscam por essa área do conhecimento (Garlet et al., 2018), ainda que segundo Neto et al. (2019), não seja raro que a escolha por uma área de conhecimento específica seja realizada sem os devidos conhecimentos das famílias e dos/as estudantes acerca do currículo e do perfil profissional do curso. No entanto, alguns direcionamentos devem ser levados em consideração, quando se pensa em processos de ensino e aprendizagem para o desenvolvimento de habilidades relacionadas a programação. Diante do exposto, compreendemos melhor quais são as reais necessidades levantadas para efetividade desse aprendizado a partir das contribuições dos autores Gomes (2010), Zanini (2012), Farias (2018) e Silva e Trentin (2016) sobre o assunto.

Conforme aponta Gomes (2010), as disciplinas de programação ganham reputação de serem difíceis, levando os alunos a terem uma imagem depreciativa da lógica de programação e daqueles que dominam o assunto. Antes mesmo de cursarem as disciplinas introdutórias de programação, os alunos já possuem um preconceito com a disciplina e acreditam ser uma matéria difícil, podendo levar o estudante à desistência ao encontrar dificuldades na aprendizagem de seus conteúdos.

De acordo com Farias et al. (2018, p. 2):

A origem do problema das reprovações na disciplina de Lógica de Programação diz respeito a alunos, professores e até mesmo às metodologias utilizadas, sendo possível destacar alguns fatores que contribuem para o cenário atual: precária base lógico-matemática dos alunos; falta de dedicação aos estudos; limitações do professor; material didático de apoio ineficiente; etc.

Ainda sobre as dificuldades apresentadas pelos alunos no aprendizado de programação os autores Silva e Trentin (2016), apontam que algumas experiências demonstram a existência de grandes dificuldades em compreender e aplicar certos conceitos abstratos de programação, em termos gerais, por parte de uma percentagem significativa dos alunos que frequentam disciplinas introdutórias nesta área. Conforme Zanini et al. (2012), para muitos alunos é uma tarefa complexa realizar a construção de um algoritmo através de um enunciado, visando solucionar um problema, apresentando dificuldades em criar uma sequência de forma coerente para a resolução adequada do problema proposto.

Para Moreira et al. (2018, p. 2), as disciplinas de Lógica de Programação exigem dos alunos habilidades como interpretação e resolução de problemas, raciocínio lógico, capacidade de abstrair soluções e aplicá-las com o uso de uma linguagem de programação, dentre outros. A falta de tais habilidades implicam nas dificuldades de aprendizagem dos alunos. Mesmo sendo uma área de conhecimento desafiadora, a aprendizagem da lógica de programação traz consigo alguns benefícios, possibilitando o desenvolvimento de uma variedade de habilidades que muitas vezes estavam ocultas e que podem alavancar grandes aprendizados na vida dos alunos.

Metodologias ativas

Considerando a importância de se adotar novas metodologias e estratégias de ensino que busquem estimular a participação dos estudantes nos processos de aprendizagem, acredita-se que por meio das metodologias ativas “estudantes e profissionais deixam o papel passivo e de meros receptores de informações, que lhes foi atribuído por tantos séculos na educação tradicional, para atribuir um papel ativo e de protagonistas da própria aprendizagem” (Filatro e Cavalcanti, p. 18, 2018).

Nesta perspectiva, Barbosa e Moura discorrem que “a diferença fundamental que caracteriza um ambiente de aprendizagem ativa é a atitude ativa da inteligência, em contraposição à atitude passiva geralmente associada aos métodos tradicionais de ensino” (Barbosa e Moura, p. 55, 2013). Sendo assim, pensando no âmbito da educação profissional, podemos considerar que essa modalidade educacional

[...] oferece muitas oportunidades de aplicar metodologias ativas de aprendizagem nas diferentes áreas de formação profissional. É o caso das aulas de laboratório, oficinas, tarefas em grupo, trabalhos em equipe dentro e fora do ambiente escolar, visitas técnicas e desenvolvimento de projetos. Essas atividades tendem a ser naturalmente participativas e promovem o envolvimento do aluno no processo de aprendizagem (BARBOSA e MOURA, p. 56, 2013).

Dentre as metodologias ativas, a aprendizagem baseada em projetos possibilita aos estudantes a realização do planejamento, implementação e avaliação dos resultados alcançados. Os projetos desenvolvidos podem durar semanas ou até mesmo meses “partindo de um problema ou uma necessidade concreta para que eles construam soluções de maneira criativa e colaborativa, resultando em um produto final” (Bernardo, 2022).

Tendo em vista que “um dos pressupostos da ABProj é a consideração de situações reais relativas ao contexto e à vida, no sentido mais amplo, que devem estar relacionadas ao objeto central do projeto em desenvolvimento” (Barbosa e Moura, p. 61, 2013). De acordo

com Barbosa e Moura, essa metodologia pode ter seus projetos classificados em: Projeto construtivo; Projeto investigativo; Projeto didático ou explicativo.

A ação desenvolvida nesta pesquisa assume a classificação de Projeto construtivo, em que objetiva construir algo novo, introduzindo alguma inovação, propor uma solução nova para um problema ou situação. Possui a dimensão da inventividade, seja na função, na forma ou no processo (Barbosa e Moura, 2013).

Essa abordagem se alinha ao dou teoria-prática na construção do saber em que, segundo Fortuna (2015), “Dentro do processo pedagógico, teoria e prática precisam dialogar permanentemente, fugindo da ideia tradicional de que o saber está somente na teoria, construído distante ou separado da ação/prática” (Fortuna, p. 64, 2015). Gemignani (2012) inclui essa relação na educação profissional e tecnológica

Estes novos instrumentos técnico-pedagógicos tornam possível a participação ativa do aluno em seu processo de aprendizagem, buscando conhecimentos, articulando teoria-prática, correlacionando os seus conhecimentos e realizando reflexões críticas sobre os problemas reais que envolvem sua formação profissional, integrados às exigências do mundo do trabalho e contribuindo para desenvolvimento da sociedade, da tecnologia e da ciência (GEMIGNANI, p.10, 2012)

Essa visão de Gemignani conversa com Freire observando que “A reflexão crítica sobre a prática se torna uma exigência da relação Teoria/Prática” (Freire, p. 24, 2016). Nesse sentido, o estudante toma uma posição crítica de sua futura profissão, atividades profissionais e a sua posição na sua prática profissional.

Interdisciplinaridade

A interdisciplinaridade é uma abordagem valiosa no ambiente escolar porque permite que os alunos vejam conexões entre diferentes disciplinas e apliquem seus conhecimentos de maneiras mais significativas (Banatto, Barros, Gemeli, Lopes e Frizon, 2012). Quando aplicada às disciplinas, princípios da lógica de programação e do desenvolvimento de sistemas, proporciona uma abordagem rica e relevante para a formação dos alunos. Essas duas áreas do conhecimento estão intrinsecamente ligadas ao mundo da tecnologia da informação e, quando combinadas, proporcionam uma visão holística do processo de criação de software.

A integração destes princípios permite aos alunos compreender a relação direta entre a lógica de programação (básica) e o desenvolvimento de sistemas (aplicações do mundo real). Eles aprendem que a lógica é a base sobre a qual constroem soluções tecnológicas específicas. Além disso, esta abordagem ajuda os alunos a ver a relevância do que estão aprendendo em sala de aula.

A interdisciplinaridade incentiva os alunos a pensar criticamente (Fazenda, Tavares e Godoy, 2015), não apenas sobre como codificar, mas também sobre como projetar sistemas de forma eficaz. Isso envolve considerações de usabilidade, eficiência, segurança e escalabilidade. O trabalho em equipe é fundamental no desenvolvimento de sistemas reais, e esta abordagem incentiva a colaboração, onde cada membro aproveita sua experiência, seja em lógica de programação ou desenvolvimento de sistemas.

Projetos práticos que envolvem a criação de sistemas completos (ou partes de sistemas) são uma forma eficaz de unir essas duas disciplinas. Os alunos podem começar com desafios simples e progredir para projetos mais complexos à medida que adquirem mais conhecimento. Além das competências técnicas, desenvolvem competências transferíveis, como a resolução de problemas, o pensamento algorítmico, a tomada de decisões informadas e as competências de comunicação, que são valiosas em muitos domínios.

A integração de questões éticas e sociais relacionadas à tecnologia em discussões e projetos conscientiza os alunos sobre as implicações de suas ações como futuros desenvolvedores de sistemas. Isso inclui tópicos como privacidade, segurança cibernética e impacto na sociedade. Por fim, a interdisciplinaridade prepara os alunos para o mercado de trabalho, onde as habilidades técnicas e o trabalho em equipe são altamente valorizados, ajudando-os a prepará-los melhor para futuras carreiras em programação, desenvolvimento e desenvolvimento de software ou áreas afins.

Trabalho colaborativo

O trabalho colaborativo pode ser muito eficaz na resolução de problemas complexos, estimulando a inovação e alcançando resultados significativos. No entanto, isso também requer habilidades interpessoais, comunicação eficaz e comprometimento de todos os membros da equipe.

Vygotsky (1989 e 1998, apud Damiani, 2008) aponta que atividades colaborativas na escola propiciam o aprendizado, pois apresentam modelos de base para o comportamento e o raciocínio do indivíduo desenvolvendo habilidades que não trabalhamos na aprendizagem individualizada. Assim, o trabalho colaborativo é fundamental para o desenvolvimento das habilidades sociais, cognitivas e emocionais dos alunos.

Entre os benefícios desse tipo de metodologia para os alunos, Damiani (2008) nos apresenta a diminuição da evasão, aumento da motivação, socialização, controle de agressividade, adaptação às normas, superação do egocentrismo, aquisição de novas habilidades, melhora no rendimento escolar, desenvolvimento da comunicação pelo trabalho

com argumentação lógica e exposição de ideias, enriquecimento do repertório de pensamentos e maior compreensão dos assuntos temáticos do trabalho. Nesse contexto, podemos sintetizar as etapas que organizaram o trabalho colaborativo desenvolvido na atuação desta pesquisa:

Formação de Equipes: O professor divide a turma em equipes de três ou quatro alunos, garantindo que haja uma mistura de habilidades e níveis de experiência em programação em cada equipe.

Definição do Projeto: O professor atribui a cada equipe um projeto específico, como criar um aplicativo de lista de tarefas, um jogo simples ou um sistema de controle de estoque. Cada equipe recebe um conjunto claro de requisitos e objetivos.

Planejamento do Projeto: As equipes realizam uma reunião inicial para discutir o projeto, definir funções e responsabilidades, criar um cronograma e decidir as tecnologias a serem utilizadas.

Desenvolvimento do Aplicativo: As equipes trabalham juntas para projetar, codificar e testar o aplicativo. Eles aplicam os conceitos de lógica de programação aprendidos em sala de aula.

Revisões de Código: Os alunos revisam o código uns dos outros, identificam erros e oferecem sugestões para melhorias. Isso promove a colaboração e a melhoria da qualidade do código.

Integração de Componentes: À medida que os diferentes componentes do aplicativo são desenvolvidos, as equipes trabalham na integração de seus códigos para garantir que o aplicativo funcione como um todo.

Testes e Depuração: As equipes realizam testes extensivos em seus aplicativos para identificar e corrigir bugs. Eles também garantem que o aplicativo atenda aos requisitos originais.

Apresentações Finais: Cada equipe apresenta seu aplicativo à turma, explicando seu funcionamento, design e os desafios enfrentados durante o desenvolvimento.

Avaliação e Feedback: O projeto é avaliado com base nos critérios, como funcionalidade, qualidade do código e apresentação. Os alunos também fornecem feedback construtivo um ao outro.

O trabalho colaborativo permitiu que os alunos aplicassem seus conhecimentos teóricos de Lógica de Programação e Desenvolvimento de Sistemas em um contexto prático. Seguiremos com o detalhamento da atuação.

RESULTADOS E DISCUSSÃO

A ação

Com o objetivo de aproximar os alunos de um ambiente onde o desenvolvimento de software é realizado em equipe com funções distintas e demonstrar aos iniciantes do curso em Técnico de Informática a produção de um software com interface gráfica, optou-se por realizar uma ação interdisciplinar entre as disciplinas de Lógica de Programação e Desenvolvimento de Sistemas.

Nesse contexto, os alunos deveriam desenvolver softwares⁶ trabalhando em equipe composta por alunos do 1º e 3º anos. As equipes seriam internamente divididas em desenvolvedores *front-end*, aos alunos do 3º ano, e *back-end*, aos alunos do 1º ano.

Para as atividades, o planejamento previu temas de projetos de software que permitissem a participação dos alunos do 1º ano, que cursavam lógica de programação e estavam tendo seus primeiros contatos com uma linguagem de programação tradicional. Todos os projetos deveriam ser desenvolvidos em linguagem Python, pois as turmas envolvidas no projeto estudavam essa linguagem em comum.

Segundo Borges (2014, p. 14), Python é uma linguagem de alto nível criada em 1990 que “tem uma sintaxe clara e concisa que favorece a legibilidade do código-fonte”. O autor expõe que essa linguagem possui estruturas de alto nível, como listas e valores complexos, além de permitir que sejam adicionadas ferramentas de terceiros. Python suporta a programação em módulos e orientada a objetos, podendo ser usada também junto com outras linguagens e com softwares na criação de scripts de comando.

Nesse caso, a meta é a otimização da aprendizagem de cada um (internalização), por meio da partilha de conhecimento e vivências (externalização). As atividades são orientadas na mesma direção e os resultados são frutos das trocas e dos objetivos compartilhados. (Cavalcante, 2018)

Dos temas escolhidos para a ação, temos: Calculadora Básica, Sorteador de Números, Software para Estacionamento, Caixa de Supermercado, Software de Agendamento de Clínica de Estética, Urna Eletrônica, Software Funcionário do Mês, Caixa com Função de Parcelamento. Cada projeto deveria contar com uma interface de usuário que permitisse a entrada de dados e exibisse as informações necessárias após o processamento.

As equipes foram formadas pelo professor das disciplinas, objetivando uma melhor distribuição de alunos entre os dois anos escolares e quantidade de grupos, tendo a lista de equipes disponibilizada em formato digital. A aproximação das turmas se deu em encontros

⁶ Tenha acesso a descrição dos projetos desenvolvidos em:
<https://docs.google.com/document/d/1IZ6EcQmQXMzD-pi5I_6R8eGniHrZWmUv9Y2nzUhwbsw/edit?usp=sharing>

presenciais fora do horário escolar devido a falta de compatibilidade de horários, porém com orientação em sala em encontros presenciais junto ao professor das disciplinas em seus respectivos horários. Assim, os alunos deveriam se encontrar para a troca de informações, gerenciamento de equipe e desenvolvimento dos códigos.

Os alunos de Lógica de Programação desenvolveram sua parte em uma característica *back-end*, devido a não terem visto conteúdos sobre *Graphic User Interface* (GUI). Utilizaram de seus conhecimentos em programação estruturada para desenvolver as funcionalidades do projeto usando Funções.

Os alunos da disciplina de Desenvolvimento de Sistemas receberam múltiplas responsabilidades para o projeto. Os estudantes trabalharam sua parte em uma perspectiva de front-end, usando seus conhecimentos em GUI, utilizando a biblioteca Tkinter da linguagem Python, somando seus estudos em gestão de projetos, deveriam atuar, também, como gestores do projeto. Assim, os alunos do 3º ano, além de desenvolverem uma interface de usuário para o sistema, auxiliaram os alunos do 1º ano no desenvolvimento do código retirando dúvidas e realizando revisões, promovendo reuniões para planejamento e execução de tarefas.

O professor, como orientador da ação, optou por uma postura de observador, realizando intervenções quando necessário ou quando a iniciativa partir dos próprios alunos, para permitir que os estudantes explorassem suas habilidades criativas e de gestão de pessoas.

Para completar o ciclo de atividades do projeto, os alunos do 3º ano realizaram uma apresentação final de seus *softwares* aos colegas do 1º ano, não apenas apresentando o resultado final, mas demonstrando e explicando o código da aplicação, informando as modificações feitas no código inicial para se adequar ao código da própria GUI.

CONSIDERAÇÕES FINAIS

Em síntese, a experiência interdisciplinar entre Lógica de Programação e Desenvolvimento de Sistemas no Centro Estadual de Educação Profissionalizante Senador Jessé Pinto Freire, no segundo semestre de 2022, revelou que a introdução de metodologias ativas, como a Aprendizagem Baseada em Projetos e a aplicação de estratégias interdisciplinares e colaborativas foram eficazes na superação das dificuldades iniciais no aprendizado de programação. A integração entre teoria e prática, aliada à interdisciplinaridade, destacou a importância da conexão entre a lógica de programação e o desenvolvimento de sistemas, proporcionando uma compreensão mais profunda e aplicada dos conceitos.

A abordagem colaborativa promoveu não apenas o aprimoramento das habilidades técnicas, mas também o desenvolvimento de competências interpessoais, comunicação eficaz e comprometimento, aspectos cruciais para o sucesso em ambientes profissionais. Em resumo, a estratégia adotada demonstrou ser efetiva na preparação dos alunos para os desafios do campo da informática, consolidando a formação técnica e profissional oferecida pelo curso.

REFERÊNCIAS

BARBOSA, E. F.; MOURA, D. G. Metodologias ativas de aprendizagem na Educação Profissional e Tecnológica. B. Tec. Senac, Rio de Janeiro, v. 39, n. 2, p. 48-67, 2013.

BERNARDO, Nairim. O que é a Aprendizagem Baseada em Projetos e como ela pode ser usada na recomposição de aprendizagens. Disponível em: <<https://novaescola.org.br/conteudo/21206/o-que-e-a-aprendizagembaseada-em-projetos-e-como-ela-pode-ser-usada-na-recomposicao-deaprendizagens>>. Publicado em NOVA ESCOLA 05 de Maio de 2022.

BONATTO, A. BARROS, C. R.. GEMELI, R. A.. LOPES, T. B.. FRISON, M. D.. **Interdisciplinaridade no Ambiente Escolar**. Seminário de Pesquisa em Educação da Região Sul, IX ANPEDSUL 2012. Disponível em <<http://www.uces.br/etc/conferencias/index.php/anpedsul/9anpedsul/paper/viewFile/2414/501>> Acesso em 05 de junho de 2017.

BORGES, Luiz Eduardo. Python para desenvolvedores: aborda Python 3.3. Novatec Editora, 2014.

DAMIANI, Magda Floriana. Entendendo o trabalho colaborativo em educação e revelando seus benefícios. Educar em revista, p. 213-230, 2008.

FARIAS, Carina Machado de; OLIVEIRA, Anderson S. de; SILVA, Everton Dias de A.. Uso do Scratch na Introdução de Conceitos de Lógica de Programação: relato de experiência. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI-SBC), 26. , 2018, Natal. Anais do XXVI Workshop sobre Educação em Computação. Porto Alegre: Sociedade Brasileira de Computação, July 2018 . ISSN 2595-6175. Disponível em:<https://sol.sbc.org.br/index.php/wei/article/view/3491> . Acesso: 08 fev. 2023.

FAZENDA, I. C. A.. TAVARES, D. E.. GODOY, H. P.. **Interdisciplinaridade na pesquisa científica**. Campinas, SP: Papyrus, 2015. (Coleção Praxis).

FORTUNA, Volnei. A relação teoria e prática na educação em Freire. REBES - Rev. Brasileira de Ensino Superior, 1(2): 64-72, out.-dez. 2015 - ISSN 2447-3944.

FREIRE, Paulo. Pedagogia da autonomia: saberes necessários à prática educativa. 53ª ed. - Rio de Janeiro: Paz e Terra, 2016.

FILATRO, Andrea. CAVALCANTI, Carolina Costa. Metodologias Inov-ativas na educação presencial, a distância e corporativa. 1.ed. - São Paulo/SP: Saraiva Educação, 2018.

GARLET, Daniela. BIGOLIN, Nara Martini. SILVEIRA, Sidnei Renato. Uma Proposta para o Ensino de Programação de Computadores na Educação Básica. Resiget –Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica, v.9, n.2, 2018. Disponível em: Acesso: 08 fev. 2023.

GEMIGNANI, Elizabeth Yu Me Yut. Formação de Professores e Metodologias Ativas de Ensino-Aprendizagem: Ensinar Para a Compreensão. Revista Fronteira das Educação [online], Recife, v. 1, n. 2, 2012. ISSN: 2237-9703. Disponível em: <http://www.frenteirasdaeducacao.org/index.php/fronteiras/article/view/14>.

GOMES, Anabela de Jesus. Dificuldades de aprendizagem de programação de computadores: contributos para a sua compreensão e resolução. Coimbra - Portugal, 2010. Universidade de Coimbra Faculdade de Ciências e Tecnologia Departamento de Engenharia Informática. Disponível em: https://files.isec.pt/DOCUMENTOS/SERVICOS/BIBLIO/teses/Tese_Dout_Anabela-Gomes.pdf. Acesso: 08 fev. 2023.

MOREIRA, Gabriel; HOLANDA, Wallace; COUTINHO, Jarbele; CHAGAS, Ferdinandy. Desafios na aprendizagem de programação introdutória em cursos de TI da UFERSA, campus Pau dos Ferros: um estudo exploratório. Anais do ECOP/UFERSA, v. 2, n. 1, 2018. Disponível em: <https://periodicos.ufersa.edu.br/index.php/ecop/article/view/7907>. Acesso: 08 fev. 2023.

NETO, Daniel; SANTOS, Eliene; SILVA, Indaiara; FRANÇA, Ivo; SILVA, Lucas. A Evasão de Estudantes nos Cursos Técnicos da Modalidade Integrada no IFBA - Campus Jacobina. Ensino em Foco, [S.l.], v. 2, n. 4, p. 37-48, abr. 2019. ISSN 2595-0479. Disponível em: <https://publicacoes.ifba.edu.br/index.php/ensinoemfoco/article/view/500> . Acesso: 08 fev. 2023.

SEVERINO, Antônio Joaquim Metodologia do trabalho científico. 23. ed. rev. atual. São Paulo. Cortez, 2007. 304 p. il.

SILVA, Bruno Siqueira da. TRENTIN, Marco Antônio Sandini. Dificuldades no Ensino-Aprendizagem de Programação de Computadores: Contribuições para a sua Compreensão e Resolução. In: Simpósio Nacional de Ensino de Ciência e Tecnologia. Anais. Ponta Grossa: Universidade Tecnológica Federal do Paraná, 2016. Disponível em: <http://www.sinect.com.br/2016/down.php?id=3349&q=1>. Acesso: 08 fev. 2023.

ZANINI, Adriana Salvador; RAABE, André Luís Alice; SANTANA, André Luiz Maciel; VIEIRA, Marli Fátima Vick. Análise dos enunciados utilizados nos problemas de programação introdutória em cursos de Ciência da Computação no Brasil. In: Anais do XXXII Congresso da Sociedade Brasileira de Computação, XX WEI– Workshop sobre Educação em Computação. 2012. Disponível em :https://www.researchgate.net/publication/306299188_Influencia_dos_enunciados_na_resolucao_de_problemas_de_programacao_introdutoria . Acesso: 08 fev. 2023.