

EXEMPLIFY-SEE: Um Portal para Apoiar Educadores na Adoção da Aprendizagem Baseada em Exemplos em Engenharia de Software

Tiago P. Bonetti¹, Williamson Silva², Thelma E. Colanzi¹

¹Departamento de Informática – Universidade Estadual de Maringá (UEM)
Maringá – PR – Brasil

²Centro de Ciências e Tecnologia – Universidade Federal do Cariri (UFCA)
Juazeiro do Norte – Ceará – Brasil

{pg55754,teclopes}@uem.br, williamson.silva@ufca.edu.br

A disciplina de Engenharia de Software (ES) demanda que estudantes compreendam conceitos e, ao mesmo tempo, desenvolvam capacidade prática para lidar com problemas e artefatos do mundo real [Sommerville 2015]. Metodologias ativas têm sido discutidas como alternativas ao modelo expositivo, por favorecerem engajamento e aprendizagem mais profunda [Lima et al. 2021, Fonseca and Gómez 2017, Jianguo et al. 2009]. Nesse contexto, a *Aprendizagem Baseada em Exemplos* (ABE) se destaca por apoiar a construção de esquemas mentais e reduzir sobrecarga cognitiva inicial, especialmente quando baseada em *worked examples* e suas variações [Van Gog et al. 2006, Renkl et al. 2009, Sweller et al. 1998]. No ensino de ES, estudos também relatam benefícios de ABE para aprendizagem conceitual e aplicação prática, com impacto em tópicos como modelagem e outros conteúdos da área [Silva et al. 2017, Silva et al. 2019, Tonhão et al. 2021]. Apesar desses benefícios, a adoção de ABE pode aumentar o esforço docente, exigindo tempo para preparar materiais, selecionar exemplos com complexidade adequada e apoiar estudantes na compreensão do contexto do exemplo [Renkl et al. 2009]. Essas barreiras motivam a necessidade de suporte operacional que ajude educadores a planejar aulas com ABE de forma guiada e reutilizável.

Nesse contexto, propomos o **EXEMPLIFY-SEE**¹, um **portal** que operacionaliza uma abordagem estruturada para adoção de ABE no ensino de ES, com foco em apoiar o planejamento e a execução de aulas por meio de fluxos orientados, repositórios e suporte à reutilização. A ferramenta organiza a construção de *planos de aula* por meio de um *tutorial interativo* (fluxo orientado por perguntas/respostas e passos sugeridos), permitindo que docentes tomem decisões instrucionais e obtenham, ao final, uma estrutura completa do plano. Além do fluxo principal de montagem do plano, o portal incorpora funcionalidades voltadas à prática cotidiana do docente e ao reuso comunitário, como (i) criação, salvamento, edição, exclusão e exportação do plano (PDF); (ii) repositório de planos (área pessoal e comunidade) com filtros e busca, e reutilização de um plano como modelo; (iii) repositório de exemplos (corretos/errôneos) colaborativo com busca e avaliação pela comunidade; e (iv) visualização das etapas da abordagem antes do uso do tutorial. Ao estruturar o planejamento e oferecer repositórios, o portal busca viabilizar o reuso e a adaptação de exemplos e planos, alinhando-se a evidências de que o *design* de atividades com exemplos exige decisões instrucionais cuidadosas

¹<https://app.exemplify-see.com/>

[Renkl et al. 2009, Van Gog et al. 2006, Chen et al. 2020] e que portais/repositórios podem contribuir para catalogação e organização de exemplos em SEE [Tonhão et al. 2020].

A avaliação do portal *EXEMPLIFY-SEE* foi conduzida para responder a duas questões de pesquisa: **(RQ1)** em que medida educadores percebem o portal como efetivo para apoiar o planejamento de aulas com ABE em Engenharia de Software? e **(RQ2)** quais são os impactos percebidos do portal na qualidade e na consistência pedagógica dos planos de aula e materiais educacionais resultantes? incluindo percepções positivas, negativas e sugestões de melhoria. Para responder a essas questões, adotamos um estudo exploratório, conduzido remotamente, no qual educadores executam uma tarefa concreta de planejamento didático com apoio do portal e, em seguida, reportam suas percepções por meio de instrumentos quantitativos e qualitativos. Esse mesmo formato de avaliação foi aplicado três vezes, sempre com uma versão evoluída do *EXEMPLIFY-SEE*, permitindo comparar evidências ao longo de ciclos sucessivos de aprimoramento. Ao longo dos três ciclos de avaliação, participaram 20 professores de diferentes universidades brasileiras. Embora os três experimentos não tenham sido conduzidos com o mesmo grupo de docentes, houve participação recorrente de alguns professores em mais de um ciclo. Em conjunto, os participantes apresentavam perfis heterogêneos quanto à experiência docente e à familiaridade com metodologias ativas.

O protocolo segue uma sequência fixa: os participantes acessam materiais instrucionais que contextualizam a aprendizagem baseada em exemplos e apresentam o funcionamento do portal; em seguida, utilizam o sistema para elaborar um plano de aula em um tópico de Engenharia de Software, explorando o fluxo guiado e os repositórios; depois respondem a um questionário pós-tarefa baseado no *Modelo de Aceitação de Tecnologia* (TAM), com itens de utilidade percebida (PU), facilidade de uso percebida (PEU) e intenção percebida de uso (IU), em escala Likert de cinco pontos [Davis et al. 1989]; e, por fim, participam de entrevista semiestruturada para aprofundar percepções, limitações e sugestões. A análise combina síntese quantitativa das respostas do TAM e codificação das entrevistas para organizar achados em categorias como aspectos positivos, negativos e melhorias [Strauss and Corbin 1998].

De modo geral, os resultados obtidos nas avaliações sucessivas indicam percepção favorável do portal quanto à utilidade e à facilidade de uso, acompanhadas de tendência de intenção de adoção, sugerindo que o suporte estruturado do sistema contribui para reduzir esforço no planejamento e aumentar previsibilidade do processo. As percepções qualitativas destacam como aspectos positivos recorrentes a clareza do passo a passo, o detalhamento das orientações e a utilidade dos repositórios para reutilização e compartilhamento de planos e exemplos. As principais oportunidades de melhoria apontadas concentram-se em ampliar a flexibilidade do fluxo sem perder consistência pedagógica, aprimorar pontos específicos de usabilidade e refinar funcionalidades relacionadas à exportação e ao reuso, além de fortalecer mecanismos de curadoria e classificação de materiais nos repositórios.

Vídeo: <https://youtu.be/RqWmroArY5Y>

Referências

Chen, X., Mitrovic, A., and Mathews, M. (2020). Learning from worked examples, erroneous examples, and problem solving: Toward adaptive selection of learning activities. *IEEE Transactions on Learning Technologies*, 13(1):135–149.

- Davis, F. D., Bagozzi, R. P., and Warshaw, P. R. (1989). User acceptance of computer technology: A comparison of two theoretical models. *Management Science*, 35:982–1003.
- Fonseca, V. M. and Gómez, J. (2017). Applying active methodologies for teaching software engineering in computer engineering. *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, 12(3):147–155.
- Jianguo, C., Huijuan, L., Lixin, A., and Yongxia, Z. (2009). Exploring teaching methods in software engineering education. *Proceedings of 2009 4th International Conference on Computer Science and Education, ICCSE 2009*, pages 1733–1738.
- Lima, J., Alencar, F., and Santos, W. (2021). A preliminary guide for assertive selection of active methodologies in software engineering education. *Brazilian Symposium on Software Engineering*, 10:170–179.
- Renkl, A., Hilbert, T., and Schworm, S. (2009). Example-based learning in heuristic domains: A cognitive load theory account. *Educational Psychology Review*, 21(1):67–78.
- Silva, F. G., Brito, M. S., Tavares, J. V. T., and von Flach G. Chavez, C. (2019). Floss in software engineering education: Supporting the instructor in the quest for providing real experience for students acm reference format. *Proceedings of the XXXIII Brazilian Symposium on Software Engineering*, pages 234–243.
- Silva, W. A. F., Steinmacher, I. F., and Conte, T. U. (2017). Is it better to learn from problems or erroneous examples? *Proceedings - 30th IEEE Conference on Software Engineering Education and Training, CSEE and T 2017*, 2017-Janua:222–231.
- Sommerville, I. (2015). *Software Engineering*. Pearson Education.
- Strauss, A. and Corbin, J. (1998). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. SAGE Publications.
- Sweller, J., Van Merriënboer, J., and Paas, F. (1998). Cognitive architecture and instructional design. *Educational psychology review*, 10(3):251–296.
- Tonh  o, S., Colanzi, T., and Steinmacher, I. (2021). Using real worked examples to aid software engineering teaching. In *Brazilian Symposium on Software Engineering*, pages 133–142. ACM.
- Tonh  o, S., Colanzi, T. E., and Steinmacher, I. (2020). A portal for cataloging worked examples extracted from open source software. In *Proceedings of the 34th Brazilian Symposium on Software Engineering*, pages 493–498, Natal Brazil. ACM.
- Van Gog, T., Paas, F., and Van Merri  nboer, J. J. (2006). Effects of process-oriented worked examples on troubleshooting transfer performance. *Learning and Instruction*, 16(2):154–164. Recent Worked Examples Research: Managing Cognitive Load to Foster Learning and Transfer.