BENEFITS OF THE PROJECT-BASED LEARNING TO COPE WITH COMPUTER PROGRAMMING EDUCATION: A SYSTEMATIC LITERATURE REVIEW

Alexandre Grotta
Instituto Federal de São Paulo (IFSP); University of Sao Paulo (USP)

Edmir Parada Vasques Prado
University of Sao Paulo (USP)

Abstract: Educators have been looking for alternative computer programing teaching methods like Project-Based Learn (PjBL) toward students' benefits, when compared to traditional teaching methods and their benefits. Given this context, this systematic literature review (SLR) aims to synthesize, categorize and describe the PjBL benefits. We adopted a recognized methodology to filter the initial 457 studies. As results, we found 44 studies that reported the main PjBL benefits to the computer programming students as follows: 1) performance; 2) motivation, engagement or interest; 3) professional aspects; 4) Communication; and other six benefits that were cited twice or more.

Introduction

Computer education programs are fronting recent challenges due to high levels of students' evasion/drop rates, lack of interest by new entrants, and changes in the manner new generations are prone to learn. Part of these challenges at related to i) computer programming education, given this is a fundamental area of the computer education and ii) the computer programming teaching methods (HEA, 2017; Queirós, 2014; Silva Filho, Motejunas, Hipólito, & Lobo, 2007; Vega, Jiménez, & Villalobos, 2012) at both education levels higher and pre-higher, such as technical high schools, are subjected to these challenges (Romeike & Göttel, 2012),

Project-based Learning (PjBL) has been applied as an alternative to the traditional computer programming teaching method in order to decrease part of these challenges, due to reports of students' benefits, such as increases of motivation and grades (Dierker, Ward, Alexander, & Donate, 2017; Goulding, 2013; Kastl, Kiesmüller, & Romeike, 2016). Anyhow, we did not a find a study that consolidates all these benefits; thus, we conduct a systematic literature review (SLR) to investigate the benefits of PjBL to computer education students. This research gap led to the main SLR question: at higher and pre-higher education, what are the PjBL benefits to computer programming students?

In addition to this section, the second section defines the theoretical bases used by this research; the third section refers to the research methodology. The fourth section reports the results and considers its limitations. The last section refers to the study conclusion.

Theoretical Bases

Computer education might be seen as a continuum or a gradient, in which the teaching methods might be a blend of instructivist and constructivist teaching methods. Instructivist methods such as demonstrations and lectures are still been reported as the most preferred one by educators. On the other hand, there is the arising of constructivist methods like problem-based learning, PjBL, puzzle-based learning, among others. However, educators might not adopt a new teaching method if they did not see clear benefits on it, such as an increase at students' motivation (Biggs & Tang, 2011; Payne, 2009; Porcaro, 2011).

PjBL might be defined as the theory and the practices towards engaging students into a unique and exploratory context, driven by the resolution of problems (DeFillippi, 2001; Romeike & Göttel, 2012). PjBL is closely related to industry and marketing. Project methods and visions were created asides the educational environment such as the waterfall method. Even further, the agile project methods were created towards software development industry needs. In fact, PjBL shares some characteristic with real-world projects, like increasing the number of communication channels among people that participates on projects. Anyhow, to apply PjBL to computer education might not be a straight and forward process due to many reasons. For instance, it requires alternative evolution instruments and methods in addition to the traditional ones (Grotta & Prado, 2018).

But beyond the mandatory measure name grade (Luckesi, 2008), PjBL educators might observe at their classes auxiliary measure, which we name as benefits. These benefits are highly relevant in the international context due to its potential to predict and influence students' final behaviors and performance (Boruchovitch, 2008). The benefits might count in favour of the student's transition from the educational environment to the professional environment (Teixeira, Bardagi, & Hutz, 2007). Even benefits back to their own community might be observed (Grotta & Prado, 2018). In general, there is a relationship among motivation, engagement, and interest at the higher education, thus these factors joined together for final analysis (Boruchovitch, 2008; Stelko-Pereira, Valle, & Williams, 2015).

In light of these fundaments, we choose to research grades benefits and additional benefits as well. We select only relevant international studies which related simultaneously: computer programming education, PjBL and benefits. In order to perform this research, we choose an SLR due to its formality, reliability and repeatability characteristics, given it identifies relevant primary studies, evaluate them and interpret these data by a particular topic

of interest (Felizardo, Nakagawa, Fabbri, & Ferrari, 2017; Kitchenham & Charters, 2007). The detailed SLR process is described in the next section.

Methodology

This section presents the SLR methodology and process recommended by Felizardo et al. (2017) split into three different phases: planning, conduction, and reporting phases. Planning and conducting phases are referred by their respective subsections as follows. The reporting phase is described in the next section.

Planning Phase

This SLR is considered a cross-disciplinarity topic, given it relates both technical and educational topics, represented by computer programming and the PjBL topics respectively. Given this context, we followed two relevant recommendations found in (Petticrew & Roberts, 2006). First, the research questions were created utilizing PICOC (Population, Intervention, Comparison, Outcome, Context) criteria.

To make the SLR more extensive, we employed four different search engines: two general-purpose and two specific-purpose search enginnes, as recommended by (Felizardo et al., 2017). As general-purpose search engines, we select Scopus and ISI Web of Science. As specific-purpose search engines, we select ERIC (Education Resources Information Center) and Technology Collection, which regard the education and technological topics respectively. We choose for scope both conference and journal publications from the past five years plus publications, thus starting on Jan 2012 and ending on Aug 2017. Finally, the SLR was planned as six sequential steps as recommend by (Felizardo et al., 2017). The following subsubsections details this criteria choose by this SLR.

Keywords definition: The base keywords were found on the two major topics of this SLR: "project based learning" and "programming". The synonyms for each keyword were defined as follows. Regarding "project based learning", the synonyms were "project based teaching", "project based instruction", "project based pedagogy" based on (Hasni et al., 2016) and PjBL. The synonyms for "programming" was "software development" based on (Godse & Godse, 2007; Jain, 2004). Additionally, we limited the language to English only in order to safeguard rigor and repeatability of this SLR by future researchers across the globe, given they might not be familiar with the authors' mother language.

The exclusion criteria were: E1 – Duplicated studies; E2 – None or superficially relation to PjBL; E3 – None or superficially relation to computer programming; E4/E5 – Study relates to a lower/upper education level, such as Fundamental or Master courses; E6 – Study does not relate technology-oriented courses, such as Accounting majors; E7 – Study relates to curricula or administrative subject; E8 – Study with access denied to the researchers. The inclusion criteria were: a paper that related to both PjBL and computer programming (or software development), to higher or pre-higher education level or both, and it has no exclusion criteria;

String definition: Generic Query String: TITLE-ABSTRACT-OR-KEY (("project based learning" OR "project based teaching" OR "project based instruction" OR "project based pedagogy" OR "PjBL") AND ("programming" OR "software development")) AND (Language = "English") AND (date between "Jan 2012-01" and "Aug 2017-31")

SLR steps definition: Step 1: Remove the filter language or dates from the query string. Run it on each search engines and memo result; Step 2: Run the *query* string (with filters) on each search engines and export the final results as Bibtex format; Step 3: Import data from step 2 into StArt and remove duplicated studies; Step 4: Remove studies by reading the title, the abstract and the keyword based on the planned exclusion criteria; Step 5: Fully read the article and collect results into StArt extraction form. If an exclusion criterion is found during the reading, stop collect and make it as reject according to its reason; Step 6: Export, classify and write the results.

Conducting Phase

We conduct the SLR from September to November 2017, starting from 457 studies and limited to 44 studies after all filtering process. As represented on table 1, the studies received a label, composed by the letter A plus a sequential number. Studies were ordered by the article title.

Table 1

All researches accepted by this SLR

Ref.	Study	Ref.	Study
A1	(Goulding, 2013)	A23	(Richard, Shryock, & Lagoudas, 2012)
A2	(Liu & Yin, 2013)	A24	(Gestwicki & McNely, 2016)
A3	(Vega, Jiménez, & Villalobos, 2013)	A25	(Zhang & Liu, 2012)

Ref.	Study	Ref.	Study
A4	(Manogaran, 2013)	A26	(Mattei et al., 2015)
A5	(Valveny et al., 2012)	A27	(Fagerholm & Vihavainen, 2013)
A6	(Lang, 2017)	A28	(Zouganeli, Tyssø, Feng, Arnesen, & Kapetanovic, 2014)
A7	(Monett, 2013)	A29	(Karaman et al., 2017)
A8	(Romeike & Göttel, 2012)	A30	(Igaki, Fukuyasu, Saiki, Matsumoto, & Kusumoto, 2014)
A9	(Kundra & Sureka, 2016)	A31	(Sunaga et al., 2017)
A10	(Abdool & Pooransingh, 2014)	A32	(Peng, Wang, & Sampson, 2017)
A11	(Lancor & Katha, 2013)	A33	(Garneli, Giannakos, Chorianopoulos, & Jaccheri, 2015)
A12	(Frank & Roeckerath, 2016)	A34	(H. Wang, Zhou, & Wu, 2016)
A13	(Jazayeri, 2015)	A35	(Kizaki, Tahara, & Ohsuga, 2014)
A14	(Vrbík & Hodinář, 2012)	A36	(Kastl et al., 2016)
A15	(Isik, 2016)	A37	(Jimenez Lopez, Prieto Pelayo, & Ramirez Forero, 2016)
A16	(Ozdamli & Turan, 2017)	A38	(Hernandez-Barrera, 2014)
A17	(H.-Y. Wang, Huang, & Hwang, 2014)	A39	(Mahnič, 2015)
A18	(Chen, Hong, & Chen, 2014)	A40	(Corno & De Russis, 2017)
A19	(Dierker et al., 2017)	A41	(Krishnamoorthy, Appasamy, & Scaffidi, 2013)
A20	(Larraza-Mendiluze et al., 2013)	A42	(Francese, Gravino, Risi, Scanniello, & Tortora, 2015)
A21	(Wu, Cassidy, McCarthy, LaRue, & Washington, 2016)	A43	(Guzman-Ramirez & Garcia, 2013)
A22	(Morimoto, 2016)	A44	(Rothe, 2015)

Legend: Underline refers to pre-higher education.

The studies come from four different educational areas, as follows: Computers Science, Information Systems and related 57% of the studies; Engineering, 30%; pre-higher level, 13%

Results

This section describes and analyse the SLR main question in the first subsection and describes its limitation in the second subsection.

PjBL Benefits

Table 2 summarizes the SLR main question: at higher and pre-higher education, what are the PjBL benefits to computer programming students? It ranks benefits by total studies.

Table 2

Benefits of PjBL to computer education students

Ranking	Benefit	Reference	Studies		
			Total	*Higher*	*Pre-Higher*
#1	grade, performance	A1, A2, A3, A4, A5, A7, A9, A11, A12, A13, A14, A19, A15, A16, A17, A21, A23, A28,	28	21	7

Ranking	Benefit	Reference	Studies		
		A29, A31, A32, A35, A36, A37, A39, A42, A43, A44			
#2	motivation, engagement and interest	A1, A2, A4, A5, A7, A8, A9, A10, A12, A13, A14, A15, A16, A17, A19, A20, A21, A23, A24, A28, A32, A36, A40, A42, A44	24	18	6
#3	Profession, customer relationship	A10, A15, A16, A35, A37, A39, A42, A44	8	7	1
#4	communication	A9, A15, A18, A22, A29, A36 A42	7	5	2
#5	problem identification and resolution	A14, A17, A25, A37	4	2	2
	team work	A7, A9, A18, A40	4	4	0
#7	cooperation	A5, A18, A35	3	3	0
#9	collaboration;	A16, A29	2	2	0
	trust and confidence;	A9, A33	2	1	1
	critical thinking;	A19, A25	2	1	1

Legend: Underline refers to pre-higher education.

The major reported benefit were grades. Given the theoretical bases, this is relevant that formal education considers impacts on grades in order to analyze PjBL benefits. Secondly, the referred studies cited the triad [motivation, engagement, or interest] as one of the major outcomes of PjBL benefits to computer programming. This is reasonable given PjBL tries to engage and thus motivates and increase the students' interest according to the theoretical bases.

Given the close relation between PjBL and the professional environment, the third benefit related to students' professional benefits. In this case, clients that represented students' customers. Similarly, there were reported enhancements on communication, given the many communication channels created by a project, in contrast with a classical instructional class, also reported benefits to the students' communication.

There is also reports of an enhancement to the teamwork and to the problem identification and resolution, which as also very close and/or leads to other related group benefits such as cooperation, collaboration, trust, and confidence, as well as critical thinking.

Limitations and Future Studies

There are two main categories of limitations to this SLR. First, the limitations related to an SLR. It includes limitations of the data quality, in which this work was based on: the limitation of language, in this case, limited to English only; the limitation of time, in this case to the past five years; among other (Felizardo et al., 2017; Kitchenham & Charters, 2007). In order to reduce the impact of these limitations, we followed the SLR literature best practices

and recommend process as much as possible. Second, limitation on how to standardize the various aspects of benefits outcome from PjBL to computer programming education (Bell, 2016; Sampieri, Collado, & Lucio, 2006). To address these limitations, we created theoretical bases to aggregate benefits into common classes of benefits. Future studies may investigate how to create standards regarding these benefits for instance.

Conclusion

The main objective of this work was to identify the major PjBL benefits to computer programming students at both higher and pre-higher levels of education. We conducted an SLR which resulted in 44 relevant studies published between Jan 2012 and Aug 2017. The first main benefits reported were improvements to students' grades performance. Secondly, students' motivation – and its related benefits interest and engagement. Other benefits were, according to the most cited ones: (#3) Profession; (#4) Communication (#5) problem identification and resolution and teamwork; (#7) Cooperation; and finally (#9) collaboration, trust and confidence and critical thinking.

This research also indicates that both higher and pre-higher education level have many similarities, including benefits to students' grades and motivation, among other common positive impacts. These findings may help educators, education leaders and related person to identify the benefits of PjBL to computer programming students, at both higher and pre-higher education levels.

References

Abdool, A., & Pooransingh, A. (2014). An Industry-Mentored Undergraduate Software Engineering Project. *2014 IEEE FRONTIERS IN EDUCATION CONFERENCE (FIE), 44.* Recuperado de https://www.computer.org/csdl/proceedings/fie/2014/3922/00/07044180-abs.html

Bell, J. (2016). Abordagens de Pesquisa. In *Projeto de Pesquisa: guia para pesquisadores iniciantes em educação, saúde e ciências sociais.* (4º ed, p. 21–22). São Paulo: Artmed Editora.

Biggs, J. B., & Tang, C. S. (2011). Effective teaching and learning for today's universities. In *SRHE and Open University Press Imprint. Teaching for quality learning at university* (4º ed, p. 1–16). New York, USA: McGraw-Hill Education.

Boruchovitch, E. (2008). Escala de Motivação Para Aprender de Universitários: Propriedades Psicométricas. *Avaliação psicológica*, *7*(2), 127–134.

Chen, C.-Y., Hong, Y.-C., & Chen, P.-C. (2014). Effects of the Meetings-Flow Approach on Quality Teamwork in the Training of Software Capstone Projects. *IEEE Transactions on Education*, *57*(3), 201–208. https://doi.org/10.1109/TE.2014.2305918

Corno, F., & De Russis, L. (2017). Training Engineers for the Ambient Intelligence Challenge. *IEEE Transactions on Education*, *60*(1, SI), 40–49. https://doi.org/10.1109/TE.2016.2608785

DeFillippi, R. J. (2001). Introduction: Project-Based Learning, Reflective Practices and Learning. *Management Learning*, *32*(1), 5–10. https://doi.org/10.1177/1350507601321001

Dierker, L., Ward, N., Alexander, J., & Donate, E. (2017). Engaging Underrepresented High School Students in

Data Driven Storytelling: An Examination of Learning Experiences and Outcomes for a Cohort of Rising Seniors Enrolled in the Gaining Early Awareness and Readiness for Undergraduate Program (GEAR UP). *Journal of Education and Training Studies*, *5*(4), 54–63.

Fagerholm, F., & Vihavainen, A. (2013). Peer Assessment in Experiential Learning Assessing Tacit and Explicit Skills in Agile Software Engineering Capstone Projects. *In: 2013 IEEE FRONTIERS IN EDUCATION CONFERENCE, 2013, New York*. Oklahoma City: IEEE.

Felizardo, K., Nakagawa, E., Fabbri, S., & Ferrari, F. (2017). *Revisão Sistematica da Literatura em Engenharia de Software*. Rio de Janeiro: Elsevier.

Francese, R., Gravino, C., Risi, M., Scanniello, G., & Tortora, G. (2015). Using Project-Based-Learning in a mobile application development course—An experience report. *Journal of Visual Languages & Computing*, *31*, 196–205. https://doi.org/http://dx.doi.org/10.1016/j.jvlc.2015.10.019

Frank, M., & Roeckerath, C. (2016). Augmenting Mathematics Courses by Problem-Based Learning. *INTERNATIONAL JOURNAL OF ENGINEERING PEDAGOGY*, *6*(1), 50–55. https://doi.org/10.3991/ijep.v6i1.5368

Garneli, V., Giannakos, M. N., Chorianopoulos, K., & Jaccheri, L. (2015). Serious game development as a creative learning experience: lessons learnt. *In: 2015 IEEE/ACM 4th International Workshop on Games and Software Engineering, 2015, Florence*, 36–42. https://doi.org/10.1109/GAS.2015.14

Gestwicki, P., & McNely, B. (2016). Interdisciplinary Projects in the Academic Studio. *ACM Transactions on Computing Education*, *16*(2). https://doi.org/10.1145/2732157

Godse, A. P., & Godse, D. a. (2007). An Introduction do Computer Programming. In *Computer Programming* (1º ed, p. 1–17). Pune: Technical Publications.

Goulding, T. (2013). A first semester freshman project: The enigma encryption system in C. *ACM Inroads*, *4*(1), 43–46. https://doi.org/10.1145/2432596.2432613

Grotta, A., & Prado, E. (2018). Um ensaio sobre a experiência educacional na programação de computadores: a abordagem tradicional versus a aprendizagem baseada em projetos. *In: 26º WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 2018, Natal - RN*, 191–200. Natal, RN.

Guzman-Ramirez, E., & Garcia, I. A. (2013). Using the Project-Based Learning Approach for Incorporating an FPGA-Based Integrated Hardware/Software Tool for Implementing and Evaluating Image Processing Algorithms Into Graduate Level Courses. *Computer Applications in Engineering Education*, *21*(1), E73–E88. https://doi.org/10.1002/cae.21563

Hasni, A., Bousadra, F., Belletête, V., Benabdallah, A., Nicole, M., & Dumais, N. (2016). Trends in research on project-based science and technology teaching and learning at K – 12 levels : a systematic review. *Studies in Science Education*, *52*(2), 199–231. https://doi.org/10.1080/03057267.2016.1226573

HEA, - Higher Education Authority. (2017). *A Study of Progression in Irish Higher Education*. Recuperado de http://hea.ie/assets/uploads/2017/06/A-Study-Of-Progression-in-Irish-Higher-Education-201213-201314.pdf

Hernandez-Barrera, A. (2014). Teaching introduction to robotics: Using a blend of problem- and project-based learning approaches. *In: IEEE SOUTHEASTCON, 2014, Lexington*. Recuperado de https://ieeexplore.ieee.org/document/6950686

Igaki, H., Fukuyasu, N., Saiki, S., Matsumoto, S., & Kusumoto, S. (2014). Quantitative Assessment with Using Ticket Driven Development for Teaching Scrum Framework. In A. Jalote, P and Briand, L and VanDerHoek (Org.), !*In: 36TH INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING (ICSE COMPANION), 2014, New York* (p. 372–381). https://doi.org/10.1145/2591062.2591162

Isik, M. F. (2016). Design and implementation of a training set for distributed system and mechatronic applications: project based learning. *Tehnicki vjesnik - Technical Gazette*, *23*(6), 1609–1616. https://doi.org/10.17559/TV-20141218095138

Jain, V. K. (2004). *Basic Computer Programming*. New Delhi: Pustak Mahal.

Jazayeri, M. (2015). Combining Mastery Learning with Project-Based Learning in a First Programming Course: An Experience Report. *In: 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering (ICSE), 2015, New York, vol.2*, 315–318. https://doi.org/10.1109/ICSE.2015.163

Jimenez Lopez, A. F., Prieto Pelayo, M. C., & Ramirez Forero, A. (2016). Teaching Image Processing in Engineering Using Python. *IEEE Revista Iberoamericana De Tecnologias Del Aprendizaje (IEEE RITA)*, *11*(3), 129–136. https://doi.org/10.1109/RITA.2016.2589479

Karaman, S., Anders, A., Boulet, M., Connor, J., Gregson, K., Guerra, W., … Vivilecchia, J. (2017). Project-based, collaborative, algorithmic robotics for high school students: Programming self-driving race cars at MIT. *In: PROCEEDINGS OF THE 7TH IEEE INTEGRATED STEM EDUCATION CONFERENCE, 2017, Princeton*, 195–203. https://doi.org/10.1109/ISECon.2017.7910242

Kastl, P., Kiesmüller, U., & Romeike, R. (2016). Starting out with projects - Experiences with agile software

development in high schools. In B. E. Vahrenhold J. Barendsen E. (Org.), !*In: Workshop in Primary and Secondary Computing Education, 4th* (Vol. 13-15-Octo, p. 60–65). https://doi.org/10.1145/2978249.2978257

Kitchenham, B., & Charters, P. (2007). A systematic review of systematic review process research in software engineering. *Information and Software Technology*, *55*(12), 2049–2075. https://doi.org/10.1016/j.infsof.2013.07.010

Kizaki, S., Tahara, Y., & Ohsuga, A. (2014). Software development PBL focusing on communication using scrum. *In: Proceedings - 2014 IIAI 3rd International Conference on Advanced Applied Informatics, IIAI-AAI, 2014, Kitakyushu*, 662–669. https://doi.org/10.1109/IIAI-AAI.2014.138

Krishnamoorthy, V., Appasamy, B., & Scaffidi, C. (2013). Using Intelligent Tutors to Teach Students How APIs Are Used for Software Engineering in Practice. *IEEE Transactions on Education*, *56*(3), 355–363. https://doi.org/10.1109/TE.2013.2238543

Kundra, D., & Sureka, A. (2016). An Experience Report on Teaching Compiler Design Concepts using Case-Based and Project-Based Learning Approaches. In S. and K. Kumar, V and Murthy (Org.), !*In: IEEE 8TH International Conference on Technology For Education (T4E), 2016, New York* (p. 216–219). https://doi.org/10.1109/T4E.2016.51

Lancor, L., & Katha, S. (2013). Analyzing PHP frameworks for use in a project-based software engineering course. *In: SIGCSE - Proceedings of the 44th ACM Technical Symposium on Computer Science Education, 2013, Denver*, 519–524. Denver.

Lang, G. (2017). Agile Learning: Sprinting through the Semester. *Information Systems Education Journal*, *15*(3), 14–21.

Larraza-Mendiluze, E., Garay-Vitoria, N., Martin, J. I., Muguerza, J., Ruiz-Vazquez, T., Soraluze, I., … Santiago, K. (2013). Game-Console-Based Projects for Learning the Computer Input/Output Subsystem. *IEEE Transactions on Education*, *56*(4), 453–458. https://doi.org/10.1109/TE.2013.2255877

Liu, Y., & Yin, L. (2013). A Project-based Learning Approach for Junior Windows Software Programmer. In P. Li (Org.), !*In: Proceedings of the 2013 Conference on Education Technology and Management Science (Icetms), 2013, Paris* (p. 906–909). Recuperado de https://www.atlantis-press.com/proceedings/icetms-13/7138

Luckesi, C. C. (2008). *Avaliação da Aprendizagem Escolar: estudos e proposições*. São Paulo: Cortez Editora.

Mahnič, V. (2015). The capstone course as a means for teaching agile software development through project-based learning. *World Transactions on Engineering and Technology Education*, *13*(3), 225–230.

Manogaran, E. (2013). ACt-PBL: An adaptive approach to teach multi-core computing in university education. *In: Proceedings - 2013 IEEE 5th International Conference on Technology for Education, T4E, 2013, Kharagpur*, 19–23. https://doi.org/10.1109/T4E.2013.13

Mattei, A. L. P., Cunha, A. M. D., Dias, L. A. V, Fonseca, E., Saotome, O., Takachi, P., … Rodrigues, G. G. D. O. (2015). Nanosatellite Event Simulator Development Using Scrum Agile Method and Safety-Critical Application Development Environment. In D. N. H. R. M. T. A. K. R. Y. S. K. V. D. L. A. S. F. T. C. Latifi S. Carneiro G. (Org.), !*In: Proceedings - 12th International Conference on Information Technology: New Generations, ITNG, 2015, Las Vegas* (p. 101–106). https://doi.org/10.1109/ITNG.2015.22

Monett, D. (2013). Agile Project-Based Teaching and Learning. *Proceedings of the International Conference on Software Engineering Research and Practice (SERP)*, 1. Recuperado de https://search.proquest.com/docview/1629366456?accountid=14643%0A

Morimoto, C. (2016). Improvement of IT Students' Communication Skills using Project Based Learning. *In: Proceedings of the 8th International Conference on Computer Supported Education, 2016, Rome*, 147–152. https://doi.org/10.5220/0005891501470152

Ozdamli, F., & Turan, B. (2017). Effects of a Technology Supported Project Based Learning (TS-PBL) Approach on the Success of a Mobile Application Development Course and the Students' Opinions. *TEM Journal-Technology Education Management Informatics*, *6*(2), 258–264. https://doi.org/10.18421/TEM62-10

Payne, C. R. (2009). Constructivism and Progressive Higher Education in the World of Information Technology. In *Information Technology and Constructivism in Higher Education: Progressive Learning Frameworks* (p. 1–29). Recuperado de https://ieeexplore.ieee.org/document/8001792

Peng, J., Wang, M., & Sampson, D. (2017). Scaffolding Project-Based Learning of Computer Programming in an Online Learning Environment. *In: 2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)*, 315–319. Piscataway, NJ, USA: The Institute of Electrical and Electronics Engineers, Inc. (IEEE).

Petticrew, M., & Roberts, H. (2006). What sort of studies do I include in the review? In *Systematic Reviews in the Social Sciences: A Practical Guide* (p. 43–48). Malden: Wiley.

Porcaro, D. (2011). Applying constructivism in instructivist learning cultures. *Multicultural Education & Technology Journal*, *5*(1), 39–54. https://doi.org/10.1108/17504971111121919

Queirós, R. (2014). *Innovative Teaching Strategies and New Learning Paradigms in Computer Programming*. Hershey, PA, USA: IGI Global.

Richard, J. C., Shryock, K. J., & Lagoudas, D. C. (2012). Integrating Aerospace Research Materials into a Project-Based First-Year Engineering Design Course. *In: 2012 ASEE Annual Conference, 2012, Washington*. Recuperado de http://www.asee.org/public/conferences/8/papers/4719/view

Romeike, R., & Göttel, T. (2012). Agile projects in high school computing education - Emphasizing a learners' perspective. *In: Proceedings of the 7th Workshop in Primary and Secondary Computing Education*, 48–57. https://doi.org/10.1145/2481449.2481461

Rothe, I. (2015). Work in Progress: Starter-Project for First Semester Students to Survey Their Engineering Studies. *In: Proceedings Of 2015 IEEE Global Engineering Education Conference (Educon), 2015, New York*, 1–4. New York, NY: IEEE.

Sampieri, R. H., Collado, C. F., & Lucio, P. B. (2006). Modelos de Pesquisa. In *Metodologia de pesquisa* (3a ed, p. 152–223). São Paulo: McGraw-Hill.

Silva Filho, R. L. L. e, Motejunas, P. R., Hipólito, O., & Lobo, M. B. D. C. M. (2007). A evasão no ensino superior brasileiro. *Cadernos de Pesquisa*, *37*(132), 641–659. https://doi.org/10.1590/S0100-15742007000300007

Stelko-Pereira, A. C., Valle, J. E., & Williams, L. C. A. (2015). Escala de engajamento escolar: análise de características psicométricas. *Avaliação Psicológica*, *14*(2), 207–212. https://doi.org/10.15689/ap.2015.1402.05

Sunaga, Y., Washizaki, H., Kakehi, K., Fukazawa, Y., Yamato, S., & Okubo, M. (2017). Relation Between Combinations of Personal Characteristic Types Educational Effectiveness for a Controlled Project-Based Learning Course. *IEEE Transactions on Emerging Topics in Computing*, *5*(1), 69–76. https://doi.org/10.1109/TETC.2016.2526664

Teixeira, M. A. P., Bardagi, M. P., & Hutz, C. S. (2007). Escalas de exploração vocacional (EEV) para universitários. *Psicologia em Estudo*, *12*(1), 195–202. https://doi.org/10.1590/S1413-73722007000100023

Valveny, E., Benavente, R., Lapedriza, À., Ferrer, M., Garcia-Barnés, J., & Sànchez, G. (2012). Adaptation of a computer programming course to the ESHE requirements: Evaluation five years later. *European Journal of Engineering Education*, *37*(3), 243–254. https://doi.org/10.1080/03043797.2012.678986

Vega, C., Jiménez, C., & Villalobos, J. (2012). Implementing an incremental project-based learning solution for CS1/CS2 courses. *CSEDU 2012 - Proceedings of the 4th International Conference on Computer Supported Education*, *2*, 15–27. Recuperado de https://www.scopus.com/inward/record.uri?eid=2-s2.0-84864866858&partnerID=40&md5=50a008048f1594841987ea32b5d3a8eb

Vega, C., Jiménez, C., & Villalobos, J. (2013). A scalable and incremental project-based learning approach for CS1/CS2 courses. *Education and Information Technologies*, *18*(2), 309–329. https://doi.org/10.1007/s10639-012-9242-8

Vrbík, V., & Hodinář, J. (2012). Communicative approach to teaching programming with focus on project-based learning. *Recent Patents on Computer Science*, *5*(2), 134–144.

Wang, H.-Y., Huang, I., & Hwang, G.-J. (2014). Effects of an Integrated Scratch and Project-based Learning Approach on the Learning Achievements of Gifted Students in Computer Courses. *In: 2014 IIAI 3rd International Conference on Advanced Applied Informatics (IIAI-AAI),2014, New York*, 382–387. https://doi.org/10.1109/IIAI-AAI.2014.85

Wang, H., Zhou, C., & Wu, Y. (2016). Smart Cup, Wisdom Creation: A Project-Based Learning Initiative for Maker Education. In P. Spector, JM and Tsai, CC and Sampson, DG and Kinshuk and Huang, R and Chen, NS and Resta (Org.), *IEEE 16th International Conference On Advanced Learning Technologies (ICALT), 2016, New York* (p. 486–488). https://doi.org/10.1109/ICALT.2016.113

Wu, L. L., Cassidy, R. M., McCarthy, J. M., LaRue, J. C., & Washington, G. N. (2016). Implementation and impact of a first-year project-based learning course. *In: ASEE Annual Conference and Exposition, Conference Proceedings, 2016, New Orleans, 2016-June*. New Orleans: American Society for Engineering Education.

Zhang, Y., & Liu, Y. (2012). Management enhanced double PBL based reform in advanced programming design course. *In: Proceedings of the 14th IEEE International Conference on High Performance Computing and Communications, HPCC-2012 - 9th IEEE International Conference on Embedded Software and Systems, ICESS, 2012, Liverpool*, 1658–1663. https://doi.org/10.1109/HPCC.2012.244

Zouganeli, E., Tyssø, V., Feng, B., Arnesen, K., & Kapetanovic, N. (2014). Project-based learning in programming classes – the effect of open project scope on student motivation and learning outcome. *IFAC Proceedings Volumes*, *47*(3), 12232–12236. https://doi.org/10.3182/20140824-6-ZA-1003.02412