PROJECT MANAGEMENT APPROACHES AND METHODS TO COMPUTER PROGRAMMING EDUCATION VIA PROJECT-BASED LEARNING: 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: This research systematizes the main project management approaches and methods that were associated with computer programming education via project-based learning (PjBL). A systematic literature review was conducted in four databases that resulted in 457 studies initially. Three project approaches were found: blended, agile and waterfall. Four project management methods were found: merged, by association or by combination; SCRUM; Extreme Programming (XP). Beside PjBL, three main teaching methods were found: traditional, problem-based learning and Agile Model for Projects in Computing Education (AMoPCE).

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). The project approaches, such as waterfall or agile, resulted in different methods types like PMI and SCRUM. They have been created asides the educational environment, given the project management profession as we know started around 1950, and nowadays there is no single approach or method to execute projects (Grotta, 2019).

Given this context, we did not find a study that consolidates the project approaches and methods that relate to PjBL; thus, we conduct a systematic literature review (SLR), in

which the main SLR question is: At higher and pre-higher education levels, what are the project management approaches and methods that related to computer education via PjBL, including those that mention the traditional teaching vision?

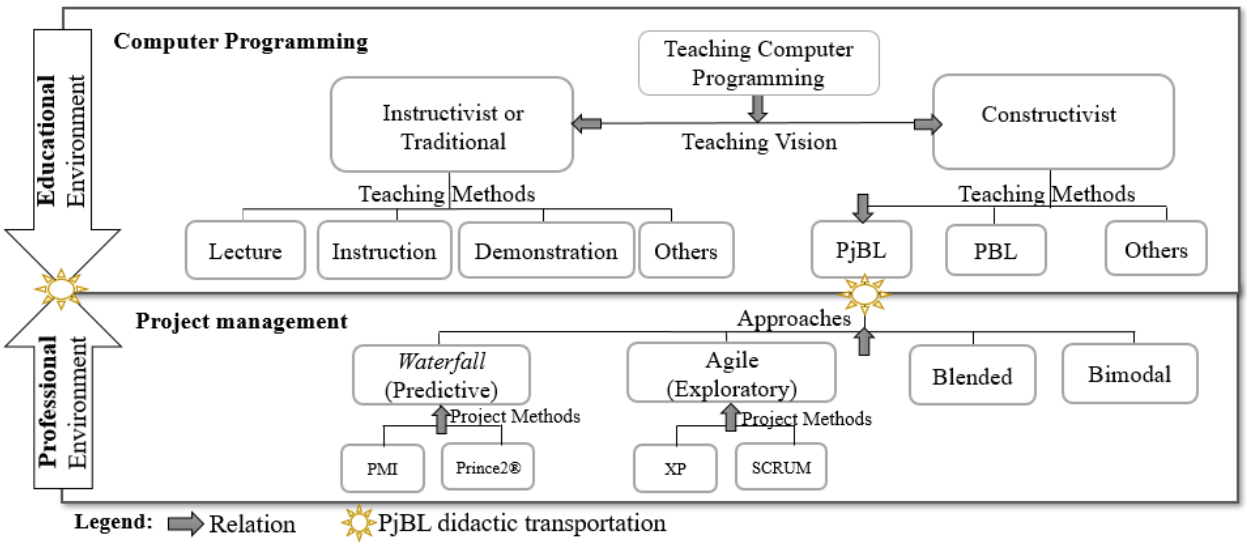
In addition to this section, the second section defines the theoretical bases used by this research; the third section refers to the research methodology. The fourth section reports the results and considers its limitations. The last section refers to the study conclusion.

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 (PBL), 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. Thus, we performed a literature review to create the theoretical bases used to moderate and classify this SLR according to figure 1 (Grotta, 2019; Grotta & Prado, 2018).

Figure 1: *Theoretical bases used to moderate and classify this RSL*



Source: adapted from (Grotta, 2019)

This figure represents PjBL as didactic transportation, given specific project approaches and their respective project methods can be adapted from a professional environment into the educational environment, such as waterfall and agile approaches. Additionally, it might be a blend of different project methods, or even a bimodal approach, which allow project managers to synchronize outcomes of both predictive and exploratory approaches. On the other hand, as seen on figure 1, there is a debate regarding the right balance of instructivist and constructivist methods, towards the most adequate to computer programming students at the educational environment (Grotta, 2019). In light of these fundaments, we choose to research these project management approaches and methods, as well as the references to the instructivist (or simply traditional teaching vision) regarding PjBL and computer programming. 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 as follows.

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 engines, 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 recommended by (Felizardo et al., 2017). The following subsubsections details these 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 in 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)
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.

Results

This section describes and analyze 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. Blended PjBL teaching method, the first occurrence on studies, were split into two types of blending: by combination, in which PjBL is mixed with other methods, as detailed in table 3; PjBL as an element used by another teaching method, as detailed in table 4. As expected by the theoretical bases, there are many references to the traditional teaching vision as seen at the #2 position. The #3 position holds all studies that made no reference to any classification used by this SLR. The #4 position holds all studies that made reference to the agile approach and/or methods, including the agile methods at positions #5 and #7. Given the relation of PBL and PjBL in theoretical bases, PBL was the #6 position.

Table 2

Project management approaches and methods related to PjBL in computer programming education

Ranking	Total	PjBL relation	Studies
#1	24	Blended PjBL teaching method	PjBL by combination: A3, A4, A9, A10, A11, A13, A16, A17, A25, A26, A31, A38, A42, A43. Total: 14. Details in table 3. PjBL as element: A1, A5, A14, A18, A24, A27, A30, A37, A40, A41. Total: 10. Details in table 4.
#2	20	Reference to the traditional teaching vision	A1, A3, A4, A5, A7, A9, A10, A13, A14, A16, A17, A19, A20, A21, A24, A25, A33, A37, A40, A43
#3	12	Non-specific PjBL	A2, A15, A19, A20, A21, A22, A23, A28, A29, A32, A33, A44
#4	10	Agile project approach	A6, A7, A8, A24, A26, A27, A30, A35, A36, A39
#5	7	Scrum project method	A8, A24, A26, A30, A35, A36, A39
#6	4	PBL teaching method	A12, A14, A34, A25
#7	2	AMoPCE teaching method	A8, A36
	2	XP project method	A7, A8
	2	Waterfall teaching method	A6, A8

Legend: Underline refers to pre-higher education.

Table 3

PjBL teaching methods by combination with other methods

ID	Teaching Method	Summary	Study
1	*Active Collaborative PBL-based teaching methodology* (ACt-PBL)	ACt-PBL is a PjBL method for teaching multiprocessing (multi-core) computing. It focuses on practical training, group and peers discussions, project demonstration, by mapping the teaching plan to the tasks assigned to the students.	A4
2	CPLC - *Case-based and Project-based Learning*	A case is a text describing a real problem, but not dealing with any solution. Based on this definition, the method proposes a main project that can be divided into subprojects and then assigned to the students.	A9
3	Double PBL	Double PBL refers to joint use the usage of both PBL and PjBL principles in order to an attempt to balance the dosages of each one.	A25
4	Interdisciplinary Project-Based Learning (IPBL)	PjBL method for interdisciplinary projects focused on nano-satellites.	A26

ID	Teaching Method	Summary	Study
5	PjBL *blended with* PBL	Similar to double PBL, this method merges the principles of PBL, but having PjBL as the mainstream.	A38
6	Customer Centric PjBL	Customer-centric PjBL was used as a blend of teaching and also for projects coming from the job market. In the end, satisfaction measurement was taken from these external customers through a specific satisfaction formula.	A35
7	Customer supported PjBL	The customer had a coaching role in the project. The customer participated in the final evaluation, the note also included the self-assessment notes of the individuals, as well as the evaluation of other groups, at the capstone project.	A10
8	PjBL with *FPGA-based tool*	Method adapted for teaching image processing using Field Programmable Gate Array (FPGA), a technology that has to gain space in research and academic environments for real-time processing.	A43
9	Person characteristics PjBL	PjBL method that considers the personal characteristics of the students using the theory of five factors and stress, in order to measure and classify the profiles of students to, among other objectives, to set up groups with better performance.	A31
10	Enhanced collaboration and competition PjBL	PjBL method where specific characteristics were included to emphasize the internal collaboration of the teams at the same time that also emphasizes the competitiveness between teams.	A42
11	PjBL *with mastery-based learning* (competence-based learning)	Teaching method that allows the student to enter the design phase of the discipline (PjBL) only after passing an optional test, called phase mastery content. If you do not pass in this first phase, the student fails at the course.	A13
12	*Scratch-oriented* PjBL	PjBL merged with Scratch language.	A17
13	Scalable and incremental project-based learning	This article is about programming course based on active learning and PjBL. It teaches by separating PROG-related domains, as opposed to focusing on the lines of code.	
14	Technology Supported Project Based Learning (TS-PBL)	PjBL for training in mobile application programming.	A16

Table 4

PjBL teaching methods that use PjBL as a component

ID	Teaching Method	Summary	Study
1	Academic studio	Method that brings together three poles (course students, educational institution and community members) around interdisciplinary learning projects. Additionally, it uses theories of sociocultural cognition.	A24
2	Team-based learning	This study presents collective (team-based) learning as an additional method of learning provided by PjBL.	A40
3	Learning by guides/samples	A technique of use of tutors, as more experienced students of other classes, to serve as guide and examples for the less experienced students.	A41
4	ARCS (Attention Relevance Confidence and Satisfaction)	According to the acronym, a method whose focus is directed by four criteria: attention, relevance, confidence, and satisfaction.	A37
5	CAT (communicative approach to teaching)	A method that focuses on students' classroom communication.	A14
	Heuristic	A more complex method than simple dialogue, it focuses on the formation of blocks of knowledge by the student, which are then revised and reused to build new blocks of knowledge.	
6	Cooperative learning	Method focusing on cooperation between students.	A5
7	meeting flow (MF)	A method that proposes the use of flexible meetings (flow of meetings), as opposed to fixed meetings proposed by agile methods, such as the daily scrum meeting.	A18

ID	Teaching Method	Summary	Study
8	Socratic-like	Socratic Method, where the teacher uses questions, questions, aiming to extract the knowledge of the students.	A1
9	TiDD (Ticket Driven Development)	It aims to guarantee a greater students' workload balance to avoid overload at small students' group, via tickets that have the content the description and plan workload for each student in advance.	A30
10	*Self and peer assessment*	Correction and evaluation technique of students' performance via both self and peers' review over the tests/evaluations results	A27

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 processes 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 project management visions and methods that are associated with both PjBL and computer programming education, including references to traditional education. We conducted an SLR which resulted in 44 relevant studies published between Jan 2012 and Aug 2017. The most cited project method was the blended PjBL teaching method, that was split into methods: PjBL by combination with other methods and PjBL as a component of other methods. Secondly, it was found the reference to the traditional teaching vision. At the #3 position, we named all studies that made no reference to any classification used by this SLR as non-specific PjBL. At the #4 position, were all studies that made any reference to the agile approach and/or methods, including those that made specific references to agile methods such as positions Scrum at #5 position and XP or AMoPCE at #7 position. We found PBL at the #6 position. In summary, three project approaches were found: blended, agile and waterfall. Four project management methods were found: merged, by association or by combination; SCRUM; Extreme Programming (XP).

Beside PjBL, three main teaching methods were found: traditional, problem-based learning and AMoPCE.

This research also indicates that there is a huge diversity of teaching methods that are named by PjBL, in which blended methods or non-specific PjBL methods take place. It is also plausible to infer that agile methods are taking place at the second block of methods, and that waterfall PjBL is been sunset or discontinued at computer programming education. There are also several studies that make references to the traditional teaching method, which might indicate studies regarding the transition from a predominant instructivist vision to a more constructivist teaching vision at computer programming education.

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.

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. (2019). *Aprendizagem baseada em projeto ágil para educação em programação de computadores no ensino superior brasileiro* (Universidade de São Paulo). Recuperado de http://www.teses.usp.br/teses/disponiveis/100/100131/tde-24012019-212610/pt-br.php

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

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

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

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