

Self-regulated learning in higher education: strategies adopted by computer programming students when supported by the SimProgramming approach

Daniela Pedrosa^{a,b*}, José Cravino^{a,b}, Leonel Morgado^{c,d}, Carlos Barreira^e

^aResearch Centre “Didactics and Technology in Education of Trainers”, Aveiro, Portugal

^bUniversidade de Trás-os-Montes e Alto Douro, Vila Real, Portugal

^cINESC TEC, Porto, Portugal

^dUniversidade Aberta, Coimbra, Portugal

^eUniversidade de Coimbra, Coimbra, Portugal

*dpedrosa@utad.pt

Abstract

The goal of the SimProgramming approach is to help students overcome their learning difficulties in the transition from entry-level to advanced computer programming, developing an appropriate set of learning strategies. We implemented it at the University of Trás-os-Montes e Alto Douro (Portugal), in two courses (PM3 and PM4) of the bachelor programmes in Informatics Engineering and ICT. We conducted semi-structured interviews with students (n=38) at the end of the courses, to identify the students' strategies for self-regulation of learning in the assignment. We found that students changed some of their strategies from one course edition to the following one and that changes are related to the SimProgramming approach. We believe that changes to the educational approach were appropriate to support the assignment goals. We recommend applying the SimProgramming approach in other educational contexts, to improve educational practices by including techniques to help students in their learning.

Keywords

Computer science. Self-regulation of learning. Teaching and learning.

How to cite this article: Pedrosa, D., Cravino, J., Morgado, L., & Barreira, C. (2017). Self-regulated learning in higher education: strategies adopted by computer programming students when supported by the SimProgramming approach. *Production, 27*(spe), e20162255. <http://dx.doi.org/10.1590/0103-6513.225516>

1. Introduction

In higher education, computer programming courses have high rates of academic failure and see students struggling, particularly in the transition from entry-level programming to advanced programming. Many reasons are pointed in the literature, namely: teaching approaches (Gomes & Mendes, 2015), lack of motivation/involvement of students in study (Gomes & Mendes, 2015), lack of motivation/involvement of students in study (Gomes & Mendes, 2015; Kumar & Khurana, 2012; Lahtinen et al., 2005; Morgado et al., 2012; Nunes et al., 2015) and the attitudes/strategies used by students in computer programming (Gomes & Mendes, 2015). Also, after graduation, most students come to the job market lacking necessary skills to meet the expectations of employers (Kumar & Khurana, 2012), such as: teamwork and cooperation skills (Sancho et al., 2009).

Self-regulated learning (SRL) is a key element for success in higher education, because it allows students to be proactive and manage their learning and the development of life skills (Fernández et al., 2013). The application of SRL strategies typically predicts high academic achievement (Broadbent & Poon, 2015), and the self-regulatory processes can be improved with appropriate interventions (Fernández et al., 2013; Zimmerman, 2008). For example, it is recommended that teachers contribute to the development of metacognitive activities,

of skills for implementing and adapting strategies for self-monitoring, make strategic use of feedback, and promote students' development of metacognitive knowledge about academic work and task-specific strategies (Cazan, 2013).

The SimProgramming pedagogical approach (Pedrosa et al., 2016a) was applied in academic years 2012/2013 and 2013/2014, in three action-research Cycles in two intermediate programming courses from the second and third curricular years of the bachelor programmes in Informatics Engineering (IE) and Information & Communication Technologies (ICT) at the University of Trás-os-Montes e Alto Douro. The 1st Cycle (Programming Methods 3 – PM3, 2012/2013) was the original implementation of the SimProgramming approach (Pedrosa et al., 2016a). This was refined in two subsequent Cycles: the 2nd Cycle in PM4 (2013/2014) and the 3rd Cycle in PM3 (2013/2014).

Under the SimProgramming approach, students develop a problem-based learning activity within the syllabus of the respective course, with a specific set of tasks based on the conceptual foundations detailed ahead.

The objectives of this work were: to identify the strategies of self-regulated learning adopted by the students during the activity according to their perceptions, and to collect data towards understanding the role of the SimProgramming approach in the application of these strategies by the students.

We conducted semi-structured interviews with students at the end of the activity (n=38), to identify the self-regulation learning strategies that students employed in the three Cycles.

The SimProgramming approach has shown promising results in the improvement of the student's learning strategies. Also, the changes in some of the specific tasks are appropriate.

2. Background

Computer programming courses are complex (Robins et al., 2003). Students experience learning difficulties (Lahtinen et al., 2005), and lack motivation and involvement in study (Morgado et al., 2012; Nunes et al., 2015). In intermediate and advanced programming courses, the level of complexity is much greater than entry-level programming courses. For example, students have difficulties grasping abstract approaches to organization of data and program control such as architectural styles, Model-View-Controller (MVC) being one of the most common, and other software engineering concepts (Cagiltay, 2007; Morgado et al., 2012), or dealing with the context of Web programming, where code is neither written nor executed in a single location, but rather distributed between the server and the client, and applications need to operate over the HTTP protocol which was designed to be stateless, hence encumbering applications with kludge-like solutions such as passing around session data in cookies or address parameters, among other difficulties (Liu & Phelps, 2011).

In learning computer programming, students that apply SRL and metacognitive strategies have good performance (Falkner et al., 2014). However, most students in computer science are not aware of SRL and metacognitive strategies, and it is necessary to infuse them in this context (Alharbi et al., 2014).

SRL is considered a meta-process that depends on the active participation of students developing academic skills, in the selection of learning strategies when conducting an academic assignment (Clark, 2012). Students demonstrate their proactive competences, monitoring and adapting the learning processes, for the regulation of metacognitive, cognitive, motivational, behavioral and environmental strategies for achieving personal goals (Zimmerman & Schunk, 2007). Interaction between compromise, self-control, autonomy and students' self-discipline allows them to regulate their actions, in order to achieve their learning goals (Hattie & Timperley, 2007).

Self-regulated learners are characterized by the fact that they construct their own meanings, goals, and strategies from the information available in the external environment and in their own minds (Pintrich, 2004). Their mastery of self-efficacy and self-knowledge makes use of various learning strategies (Zimmerman, 2013) and of the acquisition of effective practices for their study (Clark, 2012; Nicol & Macfarlane-Dick, 2006).

SRL strategies (SRLS) refers to specific skills that are part of the SRL process, and can be taught to students for they to apply in real contexts (Zimmerman, 2013). These include: strategies for goal setting and planning, organizing and transforming, seeking information, rehearsing and memorizing, environmental strategies/structuring, seeking social assistance, awareness about self-consequences, keeping records and monitoring, reviewing records, and self-evaluation (Zimmerman, 2013; Sitzmann & Ely, 2011). The adoption of SRLS helps students obtain and retain knowledge about the adoption of a methodological approach and structure their learning, affecting the results of student learning (Broadbent & Poon, 2015). According to Wang et al. (2013), the application of SRLS is usually a predictor of good academic performance.

It is recommended for the educators (teachers and tutors) the use the instructional support to help the students define appropriate goals and learning strategies (Wäschle et al., 2014; Panadero & Jonsson, 2013). Also, the educators need to develop the attention of students and create moments for helping them analyse

the task and design plans, such as formative assessment improve the competency of students for making decisions about how to use the learning resources during the course, and the capacity to coordinate learning skills, motivation and emotions to achieve the goals (Panadero et al., 2016). In higher education, workload has been identified as an important factor in the teaching and learning environment because of its impact in the quality of learning (Kyndt et al., 2014). According to Hagenauer & Volet (2014), the teacher-student relationship in higher education is considered one condition for successful learning and can explain and prevent the phenomenon of student drop-out.

The help seeking is considered another strategy of SRL than can improve knowledge (Karabenick & Newman, 2013) and is a cognitive skill that involves a set of actions, e.g. awareness of the need to seek help, identify the difficulties, identify the potential helpers, and establishing the required help (Hao et al., 2016). However, for help seeking to be a learning strategy, students need to have the intention to learn from the answer and not simply trying to avoid the activity altogether (Newman, 2008). The different components of motivation are related to SRLS and have a strong role in academic achievement (Mega et al., 2014), and the students used the motivational strategies to enhance or maintain their interest during the task (Panadero & Alonso-Tapia, 2014).

3. Teaching context

As mentioned above, we applied the learning assignment in two programming courses, along three action-research Cycles (see Figure 1): two in the Programming Methods 3 course (PM3, 2nd curricular year) and another in the Programming Methods 4 course (PM4, 3rd curricular year), of the bachelor programmes in Informatics Engineering (IE) and Information & Communication Technologies (ICT) at the University of Trás-os-Montes e Alto Douro.

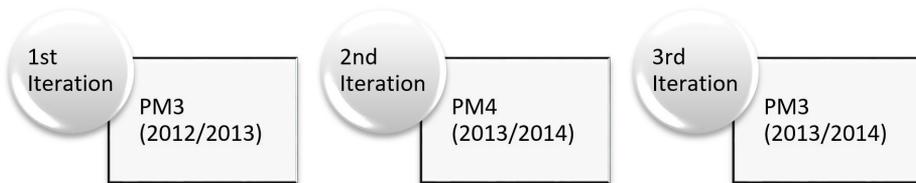


Figure 1. The three action-research Cycles.

Prior to these courses, students learned introductory programming in two previous courses, plus extra concepts in a Computational Logic course. The courses under study were provided in parallel (joint lectures, but separate hands-on lessons) to students in the IE and ICT programmes of studies. Course goals are described next.

3.1. Programming Methods 3 – PM3

The course goal is to introduce the students to large-scale programming concepts, one of the learning objective of the Association for Computing Machinery (ACM) and IEEE Computer Science Curricula (CSC) (Association for Computing Machinery, 2008). Specifically, students are introduced to the MVC architectural style, which divides programs among three blocks: the model (e.g., program state), the view (e.g., output), and the controller (e.g., program flow). The original MVC style proposal (Krasner & Pope, 1988), which handles input in the controller, is contrasted with a more recent flavour (Curry & Grace, 2008), which handles input in the view.

3.2. Programming Methods 4 – PM4

The course goal is for students to develop the knowledge and skills necessary to develop web applications. Students start by working with the client-server concept of web applications and study their operation, including analysis of the HTTP protocol and the processing of its messages by web clients and servers. The syllabus for PM4 includes data formats and metadata for web applications, including the meta-languages SGML and XML, and languages specified by them. It then proceeds with the internal operation of Web clients and servers, including automation, and concludes with the study of various types of Web applications and the specific case of Web services.

4. The SimProgramming approach and learning assignment

4.1. The SimProgramming approach

The SimProgramming approach is based on four conceptual foundations (Pedrosa et al., 2016a): 1) business-like learning environment; 2) self-regulated learning; 3) co-regulation of learning; and 4) formative assessment. Based on these, the learning activity process develops along four phases, and students have specific tasks in each phase (Pedrosa et al., 2016a).

Conceptual foundation 1: Business-like learning environment

SimProgramming stipulates the simulation of a business-like environment to promote awareness of the professional reality and expectations, through role-playing, with each participant taking on a role. The course lecturer plays the role of CEO or general manager, taking responsibility for the course content and monitoring. The others members teaching of course (e.g., course tutors or teaching assistants – henceforth “tutors”) play the role of project managers, providing close monitoring, mentoring, and feedback to students, based on the Scrum method for project management and agile software development (Paasivaara et al., 2013). Students play different roles as members of development teams.

Conceptual foundations 2: Self-regulated learning

Self-regulation learning is considered as a key aspect of motivation for active participation and involvement of students in meaningful learning activities. SimProgramming aims to promote students’ SRL strategies through active participation and engagement in meaningful activities before, during, and after completion of academic work (Wang et al., 2013).

Each student must solve his/her individual work packages and contribute to the overall perspective of the team problem. The team leader integrates research and exploration output of all members, reporting weekly at project management meetings. He/she also ensures the information flow within the team. Weekly, each student completes a self-reflection form about his/her own work, pondering on what to do the following week, and reflecting upon the factors that prevented him/her from achieving the team or individual objectives.

Students are encouraged to develop the concept of doing their work regularly and adopting study routines. This is done by creating a context where tasks are performed continuously, alongside feedback and monitoring support for self-reflection and self-regulation.

Conceptual foundation 3: Co-regulated learning

In the SimProgramming approach, the aim is to support the functional and effective development of a learning community of practices around problem solving, and students have two kinds of team tasks: reports and presentations. SimProgramming stipulates that students should be encouraged to get involved in pre-existing online communities of professionals (outside academia) not just to seek help, but to help other members of those communities, contributing to problem-solving and discussing the technologies under study or used in their future profession (Morgado et al., 2012).

Tutors (in meetings, classes, and on-line) stimulate students’ initiative when seeking social help (from peers, teachers, tutors, etc.), advising against using it as a mere least-effort approach to clarify doubts and difficulties. The tutors and the professor provide this support by advising on methods of gradual participation and involvement in communities, including suggestions of specific tasks for clarification of concepts, and advise on development of homogenous peer-based contributions and discussion, supporting community development, informal interactions, and debate, which can be promoted and monitored via a Facebook group for the course or other forms of groupware.

Conceptual foundation 4: Formative Assessment

Formative Assessment aims to improve self-reflection by providing management feedback (from tutors and from the Professor).

The Professor and tutors employ face-to-face and online contact to provide monitoring, meetings, and social media interactions, including motivational mentoring, coaching, and feedback on individual package status. The feedback is adopted as support for students’ self-regulation and critical thinking (Greene & Azevedo, 2007). Assessment in SimProgramming takes two forms: formal self-assessment of individual students and hetero-assessment by team members at the end.

4.2. Learning assignment

We combined technology-enhanced learning (Kirkwood & Price, 2014) and face to face teaching techniques to support students during the assignment (Pedrosa et al., 2016b). In both courses, problem-based learning is the approach we used: students learn while tackling a problem.

In PM3, for each team a specific problem is assigned, using a software architecture, in order to stimulate and foster advanced programming skills. Students must develop a written document with a detailed explanation of the coding approaches they used to apply an MVC-related architectural style involving different frameworks, libraries, and/or specific APIs (Morgado et al., 2012; Nunes et al., 2015; Pedrosa et al., 2016a, b).

In PM4, for each team a specific problem is assigned, using protocols, web applications, and markup languages, to develop skills on the development of web applications. Students must develop a technological solution using a web system and a given web access platform (e.g. mobile devices), and explain in detail (including code examples) how it is possible to exchange information between systems using different markup languages. (Pedrosa et al., 2016c)

Our development of the assignment, included some differences, in each one of the Cycles (see Table 1):

Table 1. Differences between the Cycles.

	PM3 (2012/ 2013)	PM4 (2013/2014)	PM3 (2013/2014)
SimProgramming Cycle	1 st Cycle, described by Pedrosa et al. (2016a)	2 nd Cycle	3 rd Cycle
Professor	Professor A	Professor B	Professor C
Type of assignment	Mandatory	Mandatory	Elective
Weight of the assignment on the final grades of students	6/20	5/20	6/20
# of tutors	N=2	N=1	N=1
Students participating in the assignment	N=97 (IE:60; TIC:37)	N=49 (EI:32; TIC:17) – All, except 4 students, participated in PM3 (2012/2013).	N=12, IE only
Students with a final grade	N=66	N=49	N=11
Established teams	N=15	N=9	N=3
Teams that completed the assignment	N=13	N=9	N=3

4.3. The SimProgramming Programming phases: learning assignment process

The learning activity develops along the semester in four phases, based in SimProgramming’s four conceptual foundations (Pedrosa et al., 2016a, b, c). In each phase, the students have different goals, and specific tasks and time frames (see Table 2).

Table 2. SimProgramming Programming phases: learning assignment process.

SimProgramming Phases	Phase 1	Phase 2	Phase 3	Phase 4
Assignment Goals	- Searching for information about the technologies under study; - Interacting in online communities; - Group work: Initiate problem-solving.	- Integration of technologies; - Group work hands-on examples.	- Improving the assignment; - Final presentation with problem-solving	- Final improvement of the assignment.
Specifics Tasks	Report interaction in community of practice (team)	Weekly forms (individual) - all phases Weekly meetings between team leaders and tutors - all phases Report interaction in community of practice (team) Report about learning progress (team) - all phases Presentation of the team work - all phases		Final Report (team) Grids about self-assessment of individual students and hetero-assessment by team members (of individuals) Extra task for extra credit or replacement credit (individuals or team)
Duration of the Phases	3 weeks	3 weeks	2 weeks	2 weeks

4.4. Changes in specifics tasks

Some specific tasks were adjusted along the research Cycles. The second Cycle took place in the 1st semester of the 2013/2014 academic year, in the PM4 course, with a different Professor (Professor B), and we did some adjustments to SimProgramming in the specific tasks (see Table 3). The third Cycle took place in the 2nd semester of the 2013/2014 academic year, in the PM3 course, with yet another Professor (Professor C). Also we changed some of the specific tasks (see Table 3).

Table 3. Specifics tasks that been changed.

	PM3 (2012/ 2013)	PM4 (2013/2014)	PM3 (2013/2014)
	Weekly forms (individual)	Fortnightly forms (team)	Fortnightly meetings between the team and tutor
Tasks	Report interaction in community of practice (team)	No reports requested	Only final report (team)
	Report about learning progress (team)		
	Final Report		
	Grids about self-assessment of individual students and hetero-assessment by team members (of individuals)	Meeting between tutor and team about self-assessment of individual students and hetero-assessment by team members (of individuals)	Meeting between tutor and team about self-assessment of individual students and hetero-assessment by team members (of individuals)

5. Methodology and data collection

We conducted semi-structured interviews (Cohen et al., 2011) about self-regulation learning strategies applied by students in the assignment. The goal of the interviews was for students to describe their self-regulation learning strategies throughout the assignment. In the three research Cycles 38 interviews were carried out: in Cycle 1, n=21; in Cycle 2, n=11; and in Cycle 3, n=6.

The student selection for the interviews was based on: role in the team (e.g. team leader), results in the assignment, special status (e.g. working-students). We also interviewed students whose assignment quality increased during the process. Some students of Cycle 1 (n=5, namely: E3; E11; E17; E18; E21) were interviewed again in Cycle 2 to identify changes in their SRL strategies.

The interview guide was constructed based on the literature (Zimmerman, 2008), and organized with five sets of questions/categories, including the one under scrutiny in this paper: *self-regulation learning strategies applied in the assignment*. In this first category *Self-regulated learning applied in the assignment*, the main questions are presented in Table 4. During the interview other issues were sometimes discussed when necessary to reach the goal.

The interviews were audio-recorded and transcribed. Interview contents were subjected to thematic analysis (Braun & Clarke, 2006). Content analysis matrixes were organized into categories, subcategories, indicators, and recording units. The categories and subcategories were developed *a priori* on the basis of the theoretical constructs. The indicators are clear definitions for each theme, which were identified in the interview responses. That is, they are words or phrases that specifically indicate the idea that the interviewee refers to, such as "Information search". Typically, each indicator represents a specific self-regulation strategy. The recording units are excerpts of the respondents' answers that illustrate particular indicators (these are numerically coded to identify the interviewees – *Exx* – and the date of the interview in the format *dd/mm/yyyy*).

Table 4. The main questions of first category.

1. What strategies did you adopt to perform the learning assignment?
2. How did you plan performing the tasks of the assignment?
3. How did you organize and plan the study for the assignment?
4. How did you manage your time?
5. Did you find the assignment difficult? Why, and how did you overcome the difficulties encountered during the assignment?
6. How did you react, when you didn't understand some concept/content related to your work?
7. How did you feel when you didn't understand some topic/content?
8. When finished the assignment, did you used any particular strategy to review the completed work?
9. And when you did the review, was it hard to do?
10. Having several academic and personal activities to deal with, how did you keep motivated and managed to accomplish the tasks?
11. Did you have a specific place for studying/working on the assignment?
12. What other strategies did you use to stay focused on you work?

The content analysis matrixes were built based on the literature about SRL (Zimmerman, 2008; 2013), and during the analysis changes were made according to what was identified in interviews, and we conducted a cyclical process of improvement, synthesis, and reflection. Several re-readings of the transcripts were undertaken, and the relevant extracts were selected. In a first moment, one researcher (one of the authors of this paper) created the content analysis matrixes (with the categories, the subcategories, indicators and the recording units). After this, another researcher independently validated or suggested changes to these content analysis matrixes. When changes were proposed, they were discussed later until both researchers reached an agreement. Finally, the final version of the content analysis matrixes was validated by all the researchers.

The answers for the category set “*self-regulation learning strategies applied in assignment*” were grouped into seven sub-categories (see Table 5) about SRL strategies (Zimmerman, 2013; Sitzmann & Ely, 2011).

Table 5. The seven sub-categories about SRL strategies.

Sub-categories	Definition
1. <i>Organizing, planning and transforming strategies</i>	Statements about the strategies that students adopted in the planning, organization, and processing of information for carrying out the assignment.
2. <i>Time management strategies</i>	Statements about time management strategies adopted by students.
3. <i>Identifying the difficulties in the assignment</i>	Statements about the difficulties students felt in doing the assignment.
4. <i>Strategies for resolution of difficulties</i>	Statements about the difficulties-solving strategies.
5. <i>Work reviewing strategies</i>	Statements indicating the students' efforts to re-read and undertake reviews of tasks.
6. <i>Motivational factors</i>	Statements about factors that influenced the motivation, from the students' perspective.
7. <i>Environmental structuring strategies</i>	Statements about environmental factors that influenced students' motivation, from their perspective.

The subcategories are organized by indicators about the types of strategies that student adopted, identifying difficulties and factors that they believed had influenced their motivation. After completing the content analysis matrixes, we organized and presented the results in tables, with the number of students who mentioned each indicator to identify what are the SRL strategies adopted by students.

6. Results

The results are organized in seven subsections, each about one of the seven sub-categories of SRL strategies. Each subsection has a table containing the indicators and the number of students who mentioned each indicator. This information is complemented with sample quotes from students.

Sub-category 1: *Organizing, planning and transforming strategies*

For the sub-category *Organizing, planning and transforming strategies* (see Table 6) the most common strategy was information search, as in most interviews students said that they searched for content related to their work. In Cycle 3 this was the only strategy that students mentioned.

Table 6. Organizing, planning and transforming strategies.

Indicators	Interviewed students		
	Cycle 1 (N=21)	Cycle 2 (N=11)	Cycle 3 (N=6)
6.1. Organizing - Information search	16	9	4
6.2. Organizing - Collected information	2	0	0
6.3. Planning - Work plan development	6	2	0
6.4. Planning - Following guidelines provided by tutors and teachers	3	2	0
6.5. Had no planned strategy	6	0	0
6.6. Transforming - Drafting notes about collected information	2	0	0
6.7. Transforming - Application of existing knowledge about the practice	1	0	0

Another strategy mentioned was following guidelines provided by the teaching team, two examples:

[...] I used it as a kind of check point [...] for the next week was a new challenge. (E18a, 29/05/2013).

[...] It was following what you asked us [...]. (E4, 20/05/2013).

We did find that some students (n=8) said they had prepared a preliminary work plan, for example:

[...] is just beginning to trace a plan [...]. (E24, 18/12/2013).

Only in Cycle 1 did some students (n=6) report not having any work planning strategy, sample quote

[...] there was no stipulated plan [...] things were done when I could [...]. (E18a, 29/05/2013).

Other strategies were mentioned less often. For example, a few students explained what they did after retrieving information (information processing, information organization, and transformation or applicability of the information).

Sub-category 2: *Time management strategies for the assignment*

In the sub-category *Time management strategies for the assignment* (see Table 7), students mentioned several strategies, without any single one standing out.

Table 7. Time management strategies.

Indicators	Interviewed students		
	Cycle 1 (N=21)	Cycle 2 (N=11)	Cycle 3 (N=6)
7.1. Lack of time	3	2	0
7.2. Lack of time due to other responsibilities	1	0	0
7.3. Initiating the activity at the last moment (procrastination)	6	2	0
7.4. Organization on weekends	1	1	0
7.5. Submitted in next week	1	0	0
7.6. Working on the assignment tasks at night	1	0	1
7.7. Grade impact factor of the assessment versus other courses	1	0	0
7.8. Division of tasks in the timeline	1	1	0
7.9. Set aside one day per week for resting	1	0	0
7.10. Placed extra activities aside	0	2	0
7.11. Working every week	2	3	0
7.12. Set aside a specific day to work on the assignment	3	1	1

Procrastination was mainly mentioned in Cycle 1, decreasing in Cycle 2. Students explained that feeling the pressure (to deliver before the deadline) forced them to try to accomplish the task, for example:

It was difficult [...], but there we go, it was always done at the last minute when the deadlines were upon us. (E23, 16/12/2013).

In Cycle 3, the students did not mention procrastination.

A few students explained that they worked during the week, and mentioned setting aside only one day per week.

They also explained that they felt unable to manage their time because of other duties or responsibilities, as seen in these two examples:

We have other courses to study. (E1, 20/05/2013).

[...] we have other things, other responsibilities. (E5, 21/05/2013).

Sub-category 3: *Identifying the difficulties in the assignment*

As shown on Table 8, in the sub-category *difficulties students encountered while performing the assignment*, these were at the level of the understanding of the theoretical content and practical implementation of the assignment.

Table 8. Identifying the difficulties in the assignment.

Indicators	Interviewed students		
	Cycle 1 (N=21)	Cycle 2 (N=11)	Cycle 3 (N=6)
8.1. Difficulties - Theoretical knowledge about the technology being studied	0	5	0
8.2. Difficulties - More than the previous year	0	1	0
8.3. Difficulties - The practical component implementation	0	1	1
8.4. Difficulties - Didn't find it complex to perform	5	4	3
8.5. Difficulties - Didn't experience any due to previous experience in PM3	0	1	0

Mostly in Cycle 2, students (n=5) specifically mentioned several difficulties, as seen in this example:

Part of it was more complicated and depended not so much the architectural style but on that part of the implementation that I did not know how to do. We do not have many bases [of knowledge]. (E32, 22/05/2014).

The difficulties expressed by students were at the individual level, but when asked about difficulties felt by their team they mentioned several. Some said that they hadn't experienced any difficulties carrying out the assignment (n=12).

Sub-category 4: *Strategies for resolution of difficulties*

As shown on Table 9, the strategy most mentioned by students to resolve their difficulties was the search for information (in Cycle 1, n=15; in Cycle 2, n=5; in Cycle 3, n=4), for example:

Table 9. Strategies for resolution of difficulties.

Indicators	Interviewed students		
	Cycle 1 (N=21)	Cycle 2 (N=11)	Cycle 3 (N=6)
9.1. Seeking Social Assistance (SSA)– Did not seek any	2	0	0
9.2. SSA - Did not seek help from teacher	5	2	2
9.3. SSA - Did not seek help from peers	5	1	2
9.4. SSA - Did not seek help due to different work pace (worker-student)	1	0	0
9.5. Resolution of Difficulties (RD)– Information search	15	5	4
9.6. RD - Use of practical exercises	1	1	2
9.7. SSA - Teachers	12	3	0
9.8. SSA - Team peers	10	6	1
9.9. SSA - Senior colleagues	3	1	2
9.10. SSA - Family member	1	0	0
9.11. SSA - Others	1	1	0
9.12. RD - Gave up seeking help, even after finding information from searching	2	2	0

Because – in theory – Google is our best friend. One has a doubt, places the question there and it answers. (E7, 21/05/2013).

This was followed by seeking social assistance from both teacher (in Cycle 1, n=12; in Cycle 2, n=3) and peers (in Cycle 1, n=10; in Cycle 2, n=6).

However, several Cycle 1 students (n=13) reported not having sought help: 2 students did not seek any help, 5 students did not seek help from the teacher and 5 students did not seek help from their peers. Also, in Cycle 3, 4 students mentioned not seeking help: 2 students did not seek help from the teacher, and 2 others did not seek help from their peers. Some reasons pointed out for not seeking the teacher were feelings of shyness, shame, fear, or inferiority.

I was shy, I was afraid that my questions were so silly that the teacher would laugh at my questions [...]. (E22, 12/12/2013).

There were students who gave up trying to solve the difficulties (in Cycle 1, n=2; in Cycle 2, n=2), i.e. they did not clarify their doubts/problems, as seen in this sample quote:

I despaired ... but progressed without solving [the doubts/problems]. (E28, 06/02/2014).

In Cycle 2 students mentioned seeking social assistance and they did not mention the opposite (not seeking it, as had happened in Cycle 1). Working-students explained their difficulties about seeking help from colleagues as being due to differences in work patterns. For example:

And they also work at a different pace from me, I work more at night, while they do not, they work during the day [...]. (E8, 22/05/2013).

Sub-category 5: *Work reviewing strategies*

The sub-category *Work reviewing strategies of the tasks required for the assignment* (see Table 10) presents students' most common strategies, which in the three Cycles were: checking for misspellings, revising sentence construction, and correcting theoretical content. As one student put it:

Table 10. Work reviewing strategies.

Indicators	Interviewed students		
	Cycle 1 (N=21)	Cycle 2 (N=11)	Cycle 3 (N=6)
10.1. No revision made	5	0	0
10.2. Component programming practice	1	2	0
10.3. Additional information	2	1	0
10.4. Typo correction and sentence construction fixes	8	2	1
10.5. Overall review and general changes	9	1	2
10.6. Spelling or grammar errors and content corrections	6	4	2
10.7. Global content (not specific)	1	0	0
10.8. Avoiding repetition of information	1	0	1
10.9. Reflecting feedback provided by teacher to other teams	0	1	0

The review I did was more in terms of misspellings, to see if the sentences made sense, it was more in that direction. (E4, 20/05/2013).

In Cycle 1, some students stated not having reviewed their work, something which in Cycle 2 was not mentioned. A fact that emerged in Cycle 2 was a student reporting to have done the review by reflecting upon comments that were given by the teacher to other teams. In Cycle 2, the focus of the review was more on content, instead of the mostly superficial revisions of Cycle 1.

[I] Tested and reviewed the code. (E27, 28/01/2014).

Sub-category 6: *Factors influencing the motivation*

For the sub-category *Factors influencing the motivation* (see Table 11), students expressed factors they felt had affected their motivation, either positively or negatively (lack of motivation).

Table 11. Factors influencing the motivation.

Indicators	Interviewed students		
	Cycle 1 (N=21)	Cycle 2 (N=11)	Cycle 3 (N=6)
11.1. Finish the programme of studies	6	2	0
11.2. Completing the PM3 or PM4 course	4	1	4
11.3. Maintaining their scholarship fund	1	0	0
11.4. Showing the ability to complete the activity	1	0	0
11.5. Achieving good results	3	1	1
11.6. Interest in programming	2	1	0
11.7. Comply with a requirement	1	2	1
11.8. Grade impact of the assignment	17	8	4
11.9. To present work that he/she agreed to do	1	0	0
11.10. Throwback to previous life experience	1	0	0
11.11. Responsibility for teamwork	1	0	0
11.12. Found the process interesting (SimProgramming approach)	1	0	0
11.13. Learning	3	3	2
11.14. Preparing for the labour market	2	2	0
11.15. Avoiding the stress of procrastination	1	0	3
11.16. Working in a new team	0	1	1
11.17. Assignment helps develop programming skills	0	9	5
11.18. Assignment helps develop interpersonal skills	0	4	0
11.19. Assignment helps the PM3/PM4 course overall	0	3	2
11.21. Lack of motivation - Is tired (of studying)	1	0	0
11.22. Lack of motivation - Overall grades are not enough to complete the course	1	1	1
11.23. Lack of motivation - Didn't learn anything new or useful	5	0	1
11.24. Lack of motivation - Wanted to learn but could not	1	0	0

Most are of a personal nature, but some are linked to interpersonal and social dimensions, such as:

The completion of the programme of studies (here “burn the ribbon” refers to a tradition of Portuguese graduating students’ to literally burn a ribbon at a celebration), as seen in his sample quote:

What motivated me the most was that this is to be my last year, I want to end [the programme] this year, burn the ribbon, tell my parents that I graduated, that was what motivated me this semester. (E27, 28/01/2014).

The perception that the work contributed to learning and skill development:

I learned a new architectural style, the differences regarding MVC, the advantages. It was worth it because we learned new things. (E7, 21/05/2013).

Also, the grade impact of the assignment was the single aspect most mentioned by students has having had an influence on motivation:

Also, I only worked because of the grade. (E15, 23/05/2013).

In Cycle 2 and Cycle 3, students mentioned other factors has having influenced their motivation, including believing that the assignment helped develop interpersonal skills, programming skills, and also that it helped understand the content of course (in Cycle2, n=3; in Cycle 3, n=2):

I gained more autonomy in programming, after all this acquired knowledge. More autonomy helps a lot. (E27, 28/01/2014).

The factors mentioned by students regarding lack of motivation, namely due to having an overall grade too low to complete the course (due to grades in other components, such as written tests):

[...] I was unhappy too, because I sacrificed a lot of time in meetings and developed it [the assignment] and gave up on other courses for this. I was disappointed, I was frustrated. But what can I do? The rules are what they are, one has to comply with the rules. (E18a, 29/05/2013).

And the feeling that they didn't learn anything new or useful, in Cycle 1 and Cycle 3 (which are the editions of the PM3 course), for example:

No, it was more that the contents were not of interest for us. [...] Maybe learning more about what we were lectured would have helped and motivated us more [...] this do not motivate anyone because we are not being lectured on that matter, we are doing something else just because. (E29, 20/05/2014).

Sub-category 7: *Environmental structuring strategies*

For the sub-category *Environmental structuring strategies* (see Table 12), in Cycle 1 and Cycle 2 most students working alone said they preferred to do it at home. Other places were mentioned, in the three Cycles, such as the library or city bars.

Table 12. Environmental structuring strategies.

Indicators	Interviewed students		
	Cycle 1 (N=21)	Cycle 2 (N=11)	Cycle 3 (N=6)
12.1. The Physical Setting (TPS) - Home	16	7	0
12.2. TPS - Library	4	2	0
12.3. TPS - Multiple locations	3	2	1
12.4. Concentration/Focusing Strategy (CS) - Silence	6	1	0
12.5. CS - Studying with music	3	1	0
12.6. CS - Having some background noise	2	0	0
12.7. CS - Place without distractions	5	2	0
12.8. CS - Inconstant	2	1	0

For focusing strategies, there is a preference for quiet places. However, no explanation was given by the students on specific strategies they used to stay focused. In Cycle 3 the students mentioned less these strategies.

7. Discussion

Students mentioned applying several SRLS, such as: information search; work reviewing; time management; social seeking assistance; resolution of difficulties; and environmental structuring. However, other strategies were also mentioned, but by fewer students, namely: strategies about information processing, organization, or application; self-reflection and self-assessment; psychological strategies to improve attention; and awareness of their difficulties.

Regarding *Organizing, planning and transforming strategies*, the most mentioned indicator was *Organizing - Information search*. As some students mentioned, the instructions were given to students by setting goals, providing task guidelines, helping students in planning the activity, which is in line with the literature in that instructional support helps students' learning (Wäschle et al., 2014; Panadero & Jonsson, 2013). However, in Cycle 1 several students mentioned that they had not planned a strategy, something that in Cycle 2 was no longer found, because the students were already familiar with the approach and type of requested tasks, and they organized themselves in a better way, e.g.:

We organized ourselves better than in the previous work. Better than in PM3. (E3b, 25/02/2014).

In Cycle 3, students did not mention in detail the type of strategies they adopted. The weekly contact between the teams and a tutor helped them in the adoption of strategies over time. But it is necessary to analyse the meetings with the tutor to ascertain that the students mentioned this. We have also a limitation in the number of students (n=6) and therefore in interviews in Cycle 3.

In all Cycles (Cycle 1, 2 and 3), few *transforming strategies* were mentioned by the students. However, in Cycle 1 "Some students mentioned in the weekly forms the adoption of transformation strategies [...]" (Pedrosa et al., 2016b, p. 98) and that an instrument to help students to conduct self-reflection is essential for the improvement of the awareness of the work that the students developed. This is in line with Panadero et al. (2016) who mention the need to create moments to help the students monitoring them learning. Regarding Cycle 2 and Cycle 3, more information may be obtained by performing a triangulation of data, including: analysis of the fortnightly forms by team (Cycle 2) and the fortnightly meetings with team and tutor (Cycle 3), to assess the hypothesis that an instrument or pedagogical help from a tutor may lead to improvement of students' self-awareness.

For *time management strategies*, procrastination was mainly mentioned in Cycle 1, but decreased in Cycle 2 and in Cycle 3 was not mentioned at all. We consider that in Cycle 2 this decrease was due to the prior experience that the students had in Cycle 1 (PM3), and that was more in line with how to manage time for the tasks they had to perform. In Cycle 3, we wonder if procrastination was not reported due to the close monitoring by a tutor. Some students mentioned devoting one day per week to carry out the tasks of the activity. The guidelines may have led students to organize their schedule according to deadlines, an impact of instructional support which was reported by Wäschle et al. (2014).

In all three Cycles, students mentioned several reasons for the lack of time to manage their tasks: "[...] because of tasks and tests they need to account for in the various courses throughout the semester" (Pedrosa et al., 2016b, p. 98), confirmed the impact of workload in learning (Kyndt et al., 2014). Therefore, careful coordination between the professors of the various courses is essential in the design of goals and learning processes, to avoid causing an overload of work for students.

Regarding *Identifying of the difficulties in the assignment*, in all three Cycles some students (n=12) mentioned that the activity was not too complex. In Cycle 2, students mentioned difficulties felt both at the level of theoretical understanding and practical accomplishments. In Cycle 1, we found that students specified their difficulties in individual forms, which lead us to think that this is a helpful instrument to make students aware of their specific difficulties in the tasks (Pedrosa et al., 2016b).

Regarding *strategies for resolution of difficulties*, the most mentioned strategy was the search for information, but also mentioned was the seeking of social assistance from teacher and peers. However, several Cycle 1 students reported not having sought help. Some reasons pointed out for not seeking the teacher were feelings of shyness, shame, fear, or inferiority. There were students who gave up trying to solve the difficulties, i.e., did not clarify their doubts/problems, confirming the complexity and importance of teacher-student relationships (as reported by Hagenauer & Volet, 2014). Also, we know that in Cycle 1, students (although unsuccessfully) sought help in online communities, which is one of the tasks required by Simprogramming approach (Pedrosa et al., 2016a, b), confirming that the students need to have intention for learning with the answer of (Newman, 2008).

As for *work review strategies*, the most common were reviewing spelling and grammar errors, content corrections, and sentence construction fixes. In Cycle 2 and Cycle 3, the focus of the review was more on content, instead of the mostly superficial revisions of Cycle 1. Some students from Cycle 1 stated not having reviewed their work at all, something which in Cycle 2 and Cycle 3 was not mentioned.

The students expressed factors that affected their motivation, most are personal in nature but the interpersonal and social aspects are also mentioned. In the Cycle 2 and Cycle 3, students were aware of the importance of work for programming and interpersonal skills.

For *environmental structuring strategies*, students mentioned various locations, however, for the concentration/focusing strategies, no explanation was given by the students on specific strategies. This lack of concentration strategies by students is worrying.

Throughout the discussion, we have been referring to the importance of appropriate teaching techniques (instruction, tutor monitoring) as aspects that have impact in the improvement of learning for students in various levels of the SRLS. The SimProgramming approach encourages the appropriate development of strategies and teaching techniques to help students achieve the learning objectives.

We found some improvements in strategies from Cycle 1 to Cycle 2, in students who participated in the activity in both PM3 (Cycle 1) and PM4 (Cycle 2). We consider that the experience of having already conducted an assignment within the SimProgramming approach helped carrying out the activity in PM4 (Cycle 2). Also, some changes to specific tasks have shown to be adequate, as explained by a student:

[...] in the last year [Cycle 1 PM3 – 2012/2013] they changed to that [approach] [...] [which] improved and we became more interested [...] And that was good because it was an approach that we were now used to [...] already had work experience with and used it this year [...]. (E26, 27/01/2014).

8. Conclusions and future work

Students mentioned various SRL strategies, namely organizing, planning and transforming strategies. However, they did not mention in detail how they process and work the information they collect (transforming strategies). In terms of time management strategies, students pointed out the factors that made it difficult to manage their time, such as workload.

When asked to identify the difficulties they experienced in the assignment, students did not mention in detail the difficulties they felt, and some said they felt no difficulty. The strategies for solving the difficulties varied for each student. We had students who sought social assistance (teachers and peers), as well as students who did not look for it. The most adopted strategy was to search for information.

As for work review strategies, essentially global review strategies were mentioned, with little reference to further review strategies.

Motivational factors varied, most of them related with learning and an opportunity for the development of programming and interpersonal skills, the grade impact of the assignment, and the opportunity to complete the course. However, we also had students mentioning aspects that discouraged them.

In terms of environmental structuring strategies, as for the physical organization several places were referred. On the other hand, little psychological organization strategies were mentioned by the students.

The results are encouraging regarding the ability of the SimProgramming approach to help improve students' self-regulated learning strategies, namely: information search; work reviewing; time management; social seeking assistance; resolution of difficulties; and environmental structuring. These strategies are related with the intended objectives of the SimProgramming approach (for example, in Phase 1, one of the assignment goals is searching for information on the technologies under study). However, it is necessary to adopt teaching techniques that stimulate other strategies, for example: strategies about information processing, organization, or application; self-reflection and self-assessment; psychological strategies to improve attention; and awareness of one's own difficulties.

Participation in PM3 (Cycle 1) activities using the SimProgramming approach helped students develop the self-regulation learning strategies that later emerged in Cycle 2, because improvements from Cycle 1 to Cycle 2 were indeed observed in some SRLS. For example, the improvement of self-awareness may be due to the previous experience students had during PM3 (Cycle 1). It is feasible that the previous experience with the SimProgramming approach (for Cycle 2) and the close monitoring by tutor (in Cycle 3) helped students stay more aware of the need to review the tasks.

Across the three Cycles, results do not contradict the notion that the SimProgramming approach supports self-regulated learning, namely, the improvement of the student's learning strategies. The changes that have been made over time to the SimProgramming approach, particularly in some of the specific tasks, are thus deemed worthy of further analysis in subsequent research efforts. The development of activities to raise awareness in students about the various types of strategies that can be pursued for success in academic tasks may be a promising path for improving the approach further.

There are some limitations to this study because it was developed in a specific context and with short sample sizes (mainly in Cycle 3). To allow for further generalization, it is necessary to develop follow-up studies about the application of the SimProgramming approach in other pedagogical contexts, with the appropriate adaptation of the tasks, as well as the adoption of activities that help the students to develop the strategies that were few mentioned. Also, specific data gathering and controls should be planned to provide more insights on the relationship between outcomes and specific features of the approach and its deployment. For instance, future

research about the SimProgramming approach should strive to achieve a better understanding of the impact of team work and assignment grade on the self-regulation of student learning.

Acknowledgements

Pedrosa, D. thanks the Fundação para a Ciência e Tecnologia (FCT), Portugal, for Ph.D. Grant SFRH/BD/87815/2012. We would like to thank all the students and teachers who collaborated on this research.

References

- Alharbi, A., Henskens, F., & Hannaford, M. (2014). personalised learning object system based on self-regulated learning theories. *International Journal of Engineering Pedagogy*, 4(3), 24-35. <http://dx.doi.org/10.3991/ijep.v4i3.3348>.
- Association for Computing Machinery – ACM. IEEE Computer Society – IEEE-CS, 2008. *Computer science curriculum 2008: an interim revision of CS 2001* (Computing Curriculum Series). New York: ACM. Retrieved in 14 October 2016, from <http://www.acm.org/education/curricula/ComputerScience2008.pdf>
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77-101. <http://dx.doi.org/10.1191/1478088706qp0630a>.
- Broadbent, J., & Poon, W. L. (2015). Self-regulated learning strategies & academic achievement in online higher education learning environments: a systematic review. *The Internet and Higher Education*, 27, 1-13. <http://dx.doi.org/10.1016/j.iheduc.2015.04.007>.
- Cagiltay, N. E. (2007). Teaching software engineering by means of computer-game development: challenges and opportunities. *British Journal of Educational Technology*, 38(3), 405-415. <http://dx.doi.org/10.1111/j.1467-8535.2007.00705.x>.
- Cazan, A.-M. (2013). Teaching self regulated learning strategies for psychology students. *Procedia: Social and Behavioral Sciences*, 78, 743-747. <http://dx.doi.org/10.1016/j.sbspro.2013.04.387>.
- Clark, I. (2012). Formative assessment: assessment is for self-regulated learning. *Educational Psychology Review*, 24(2), 205-249. <http://dx.doi.org/10.1007/s10648-011-9191-6>.
- Cohen, L., Manion, L., & Morrison, K. (2011). *Research methods in education* (7th ed.). London: Routledge-Taylor & Francis Group.
- Curry, E., & Grace, P. (2008). Flexible self-management using the model-view-controller pattern. *IEEE Software*, 25(3), 84-90. <http://dx.doi.org/10.1109/MS.2008.60>.
- Falkner, K., Vivian, R., & Falkner, N. J. (2014). Identifying computer science self-regulated learning strategies. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education* (pp. 291-296), Uppsala, Sweden. <http://dx.doi.org/10.1145/2591708.2591715>.
- Fernández, E., Bernardo, A., Suárez, N., Cerezo, R., Núñez, J. C., & Rosário, P. (2013). Predicción del uso de estrategias de autorregulación en educación superior. *Anales de Psicología*, 29(3), 865-875. <http://dx.doi.org/10.6018/analesps.29.3.139341>.
- Gomes, A. J., & Mendes, A. J. (2015). À procura de um contexto para apoiar a aprendizagem inicial de programação. *Educação, Formação & Tecnologias*, 8(1), 13-27.
- Greene, J. A., & Azevedo, R. (2007). A theoretical review of Winne and Hadwin's model of self-regulated learning: new perspectives and directions. *Review of Educational Research*, 77(3), 334-372. <http://dx.doi.org/10.3102/003465430303953>.
- Hagenauer, G., & Volet, S. E. (2014). Teacher-student relationship at university: an important yet under-researched field. *Oxford Review of Education*, 40(3), 370-388. PMID:27226693. <http://dx.doi.org/10.1080/03054985.2014.921613>.
- Hao, Q., Barnes, B., Wright, E., & Branch, R. M. (2016). The influence of achievement goals on online help seeking of computer science students. *British Journal of Educational Technology*. In press. <http://dx.doi.org/10.1111/bjet.12499>.
- Hattie, J., & Timperley, H. (2007). The power of feedback. *Review of Educational Research*, 77(1), 16-17. <http://dx.doi.org/10.3102/003465430298487>.
- Karabenick, S. A., & Newman, R. S. (Eds). (2013). *Help seeking in academic settings: goals, groups, and contexts*. London: Routledge-Taylor & Francis Group.
- Kirkwood, A., & Price, L. (2014). Technology-enhanced learning and teaching in higher education: what is “enhanced” and how do we know? A critical literature review. *Learning, Media and Technology*, 39(1), 6-36. <http://dx.doi.org/10.1080/17439884.2013.770404>.
- Krasner, G. E., & Pope, S. T. (1988). A description of the model-view-controller user interface paradigm in the Smalltalk-80 System. *Journal of Object-Oriented Programming*, 1(3), 26-49.
- Kumar, B., & Khurana, P. (2012). Gamification in education: learn computer programming with fun. *International Journal of Computers and Distributed Systems*, 2(1), 46-53.
- Kyndt, E., Berghmans, I., Dochy, F., & Bulckens, L. (2014). 'Time is not enough.' Workload in higher education: a student perspective. *Higher Education Research & Development*, 33(4), 684-698. <http://dx.doi.org/10.1080/07294360.2013.863839>.
- Lahtinen, E., Ala-Mutka, K., & Järvinen, H. M. (2005). A study of the difficulties of novice programmers. *ACM SIGCSE Bulletin*, 37(3), 14-18. <http://dx.doi.org/10.1145/1151954.1067453>.
- Liu, Y., & Phelps, G. (2011). Challenges and professional tools used when teaching web programming. *Journal of Computing Sciences in Colleges*, 26(5), 116-121.
- Mega, C., Ronconi, L., & De Beni, R. (2014). What makes a good student? How emotions, self-regulated learning, and motivation contribute to academic achievement. *Journal of Educational Psychology*, 106(1), 121-131. <http://dx.doi.org/10.1037/a0033546>.
- Morgado, L., Fonseca, B., Martins, P., Cruz, G., Maia, A. M., Nunes, R., & Santos, A. (2012). Social networks, microblogging, virtual worlds, and Web 2.0 in the teaching of programming techniques for software engineering: A trial combining collaboration and social interaction beyond college. In *Proceedings of the Global Engineering Education Conference (EDUCON)* (pp. 1-7), Marrakech, Morocco.

- Newman, R. S. (2008). The motivational role of adaptive help seeking in self-regulated learning. In D. H. Schunk & B. J. Zimmerman (Eds.), *Motivation and self-regulated learning: theory, research, and applications* (pp. 315-338). Mahwah: Erlbaum.
- Nicol, D., & Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: a model and seven principles of good feedback practice. *Studies in Higher Education, 31*(2), 199-218. <http://dx.doi.org/10.1080/03075070600572090>.
- Nunes, R. R., Pedrosa, D., Fonseca, B., Paredes, H., Cravino, J., Morgado, L., & Martins, P. (2015). Enhancing students' motivation to learn software engineering programming techniques: a collaborative and social interaction approach. In M. Antona & C. Stephanidis (Eds.), *Universal access in human-computer interaction: access to learning, health and well-being* (pp. 189-201). Switzerland: Springer International Publishing.
- Paasivaara, M., Lassenius, C., Damian, D., Rätty, P., & Schröter, A. (2013). Teaching students global software engineering skills using distributed scrum. In *Proceedings of the 2013 35th International Conference on Software Engineering (ICSE)* (pp. 1128-1137), San Francisco, CA, USA.
- Panadero, E., & Alonso-Tapia, J. (2014). How do students self-regulate? Review of Zimmerman's cyclical model of self-regulated learning. *L'Année Psychologique, 30*(2), 450-462.
- Panadero, E., & Jonsson, A. (2013). The use of scoring rubrics for formative assessment purposes revisited: a review. *Educational Research Review, 9*, 129-144. <http://dx.doi.org/10.1016/j.edurev.2013.01.002>.
- Panadero, E., Jonsson, A., & Strijbos, J. W. (2016). Scaffolding self-regulated learning through self-assessment and peer assessment: guidelines for classroom implementation. In D. Laveault & L. Allal (Eds.), *Assessment for learning: meeting the challenge of implementation* (pp. 311-326). Springer International Publishing.
- Pedrosa, D., Cravino, J., Morgado, L., Barreira, C., Nunes, R. R., Martins, P., & Paredes, H. (2016a). Simprogramming: the development of an integrated teaching approach for computer programming in higher education. In *Proceedings of the 10th annual International Technology, Education and Development Conference (INTED 2016)*, Valencia, Spain.
- Pedrosa, D., Cravino, J., Morgado, L., & Barreira, C. (2016b). Self-regulated learning in computer programming: strategies students adopted during an assignment. In *Proceedings of the International Conference on Immersive Learning* (pp. 87-101), Santa Barbara, CA, USA.
- Pedrosa, D., Cravino, J., Morgado, L., & Barreira, C. (2016c). Self-regulated learning in higher education: strategies adopted by computer programming students. In *Proceedings of the 8th International Symposium on Project Approaches in Engineering Education (PAEE)*, Guimarães, Portugal.
- Pintrich, P. (2004). A conceptual framework for assessing motivation and self-regulated learning in college students. *Educational Psychology Review, 16*(4), 385-407. <http://dx.doi.org/10.1007/s10648-004-0006-x>.
- Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: a review and discussion. *Computer Science Education, 13*(2), 137-172. <http://dx.doi.org/10.1076/csed.13.2.137.14200>.
- Sancho, P., Moreno-Ger, P., Fuentes-Fernández, R., & Fernández-Manjón, B. (2009). Adaptive role playing games: an immersive approach for problem based learning. *Journal of Educational Technology & Society, 12*(4), 110-124.
- Sitzmann, T., & Ely, K. (2011). A meta-analysis of self-regulated learning in work-related training and educational attainment: what we know and where we need to go. *Psychological Bulletin, 137*(3), 421-442. PMID:21401218. <http://dx.doi.org/10.1037/a0022777>.
- Wang, C. H., Shannon, D., & Ross, M. (2013). Students' characteristics, self-regulated learning, technology self-efficacy, and course outcomes in online learning. *Distance Education, 34*(3), 302-323. <http://dx.doi.org/10.1080/01587919.2013.835779>.
- Wäschle, K., Allgaier, A., Lachner, A., Fink, S., & Nückles, M. (2014). Procrastination and self-efficacy: tracing vicious and virtuous circles in self-regulated learning. *Learning and Instruction, 29*, 103-114. <http://dx.doi.org/10.1016/j.learninstruc.2013.09.005>.
- Zimmerman, B. J. (2008). Investigating self-regulation and motivation: historical background, methodological developments, and future prospects. *American Educational Research Journal, 45*(1), 166-183. <http://dx.doi.org/10.3102/0002831207312909>.
- Zimmerman, B. J. (2013). From cognitive modeling to self-regulation: a social cognitive career path. *Educational Psychologist, 48*(3), 135-147. <http://dx.doi.org/10.1080/00461520.2013.794676>.
- Zimmerman, B. J., & Schunk, D. H. (2007). Motivation: an essential dimension of self-regulated learning. In D. H. Schunk & B. J. Zimmerman (Eds.), *Motivation and self-regulated learning: theory, research, and applications* (pp. 1-30). Mahwah: Lawrence Erlbaum.

Received: Oct. 14, 2016

Accepted: June 14, 2017