García-Peñalvo, F. J., Reimann, D., & Maday, C. (2018). Introducing Coding and Computational Thinking in the Schools: The TACCLE 3 – Coding Project Experience. In M. S. Khine (Ed.), Computational Thinking in the STEM Disciplines. Foundations and Research Highlights (pp. 213-226). Cham, Switzerland: Springer.
doi:10.1007/978-3-319-93566-9_11

Introducing Coding and Computational Thinking in the Schools: The TACCLE 3 – Coding Project Experience

Francisco José García-Peñalvo, Daniela Reimann, and Christiane Maday

PRE-PRINT
GRIAL

1 Introduction

There is a general trend worldwide to make computer science a basic skill (García-Peñalvo et al. 2017; Llorens Largo et al. 2017). This is related to future generations of workers that should know, at least, the basic laws of a computer-based society and, without demerit to humanities or social sciences, trying to reduce the current gap with STEM (science, technology, engineering, and mathematics) (CEDEFOP 2015) careers.

Current society is software-driven (Manovich 2013). A very common situation in countries with a high rate of unemployment is they have unfilled positions for engineers and technicians for the industry and digital services. This means that a growth in the demand of positions related to technology and scientific knowledge, particularly engineering, but not only, is not reflected in the increase of students in such university degrees.

In the European Union, more than 800,000 professionals skilled in computing/informatics by 2020 are expected; many educators, parents, economists, and politicians are starting to think that students need some computing and coding skills (Balanskat and Engelhardt 2015).

In EEUU there are different studies that recommend the creation of a well-defined set of K-12 computer science standards based on algorithmic/computational thinking concepts (Tucker et al. 2006; Wilson et al. 2010).

F. J. García-Peñalvo (✉)
Research Institute for Educational Sciences, University of Salamanca, Salamanca, Spain
e-mail: fgarcia@usal.es

D. Reimann · C. Maday
Karlsruhe Institute of Technology, Institute of Vocational and General Education, Karlsruhe, Germany

On the other hand, new devices (Alonso de Castro 2014; Sánchez-Prieto et al. 2013, 2014), from smartphones and tablets to electronic learning toys and robots, find new audiences with increasingly young children. This causes new challenges for teachers (Sánchez-Prieto et al. 2016a, b, 2017), for example, how to define developmentally appropriate activities and content for children of different ages (Bers et al. 2014).

These new devices caused that the real life in the physical space is represented in the virtual space in all its facets, e.g., the place where I am, the activities I am undertaking, with whom I communicate and interact, and what I buy. Data traces in the virtual space, which capture more and more what we do, are stored, networked, and sent to third parties (Boyd and Ellison 2008; Guettat et al. 2010). At the same time, the subjects in the digitized world always receive more accurate proposals and offers of assistance systems from the virtual space (Chajri and Fakir 2014; Colomo-Palacios et al. 2017). The virtual affects the physical reality to an increasing extent, but the virtual space is not a "neutral world," but it is driven by corporations and their business interests.

Whereas information technology (IT) literacy is the capability to use today's technology in one's own field, the notion of IT fluency adds the capability to independently learn and use new technology as it evolves (National Research Council Committee on Information Technology Literacy 1999) throughout one's professional lifetime. Moreover, IT fluency also includes the active use of algorithmic thinking (including programming) to solve problems, whereas IT literacy is more limited in scope.

The most frequent approach to teaching digital literacy has been to gradually encourage the learning of programming, and the term code literacy (Prensky 2008) has been coined to refer to the process of teaching children programming tasks, from the simplest and most entertaining to the most complex; this way the student's progress is centered on the difficulty of the tasks and in their motivating characteristic. This means a link between the learning with the response to a stimulus instead to the child's learning and cognitive capabilities, following the traditional behaviorist theories (Zapata-Ros 2015).

However, there exist an alternative in the constructionism approach, yet considered by Papert (1980) in his researches based on the Logo programming language, that conveys the idea that the child actively builds knowledge through experience and the related "learn-by-doing" approach to education. Papert wanted to create "a mathematics children can love rather than inventing tricks to teach them a mathematics they hate," because Papert's leitmotifs were thinking about thinking and the freedom to achieve one's potential (Stager 2016).

The term computational thinking was made popular by Jeannette M. Wing (2006), with her definition "computational thinking involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science." Aho (2012) simplified this concept defining it as the thought processes involved in formulating problems, so "their solutions can be represented as computational steps and algorithms." García-Peñalvo (2016f) defined computational thinking as the application of high level of abstraction and an algorithmic approach to solve any kind of problems.

PRE-PRINT GRIAL

European Erasmus+ TACCLE 3 – Coding project (García-Peñalvo 2016a, b,c, d; TACCLE 3 Consortium 2017) focusses on supporting school teachers and developing their confidence to deliver the new computing curriculum including coding and computational thinking approaches.

In this chapter, TACCLE 3 – Coding is introduced, and in this framework, the experience of using wearables with target groups in higher education (pedagogy, engineering pedagogy) as well as in elementary teacher training is going to be presented.

2 TACCLE 3 – Coding Project

TACCLE 3 – Coding is a European Union Erasmus+ KA2 Programme project that supports primary school staff and others who are teaching computing to 4–14-year-olds. It started at September 2015 and will end at October 2017.

The project consortium is coordinated by GO! Het Gemeenschapsonderwijs (Belgium) and composed of the following partners: the Pontydysgu Limited (United Kingdom), Scholengroep 1 Antwerpen (Belgium), Karlsruher Institut Für Technologie (Germany), Hariduse Infotehnoloogia Sihtasutus (Estonia), Tallinn University (Estonia), University of Salamanca (Spain), Aalto-Korkeakoulusaatio (Finland), and Itä-Suomen yliopisto (Finland).

All the information and the project outcomes and deliverables are available at the project website http://www.taccle3.eu, and they are licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

Also, training courses will be available for both in-service and future teachers.

Many European countries are introducing computing and coding as core curriculum topics (Balanskat and Engelhardt 2015). Some have already done so; many others are intending to. Inevitably the detail of the curricula will be different in each country, but there is a substantial overlap – almost all of the curricula available so far include programming, control technology, and computational/logical thinking, so TACCLE 3 has started with these (García-Peñalvo et al. 2016).

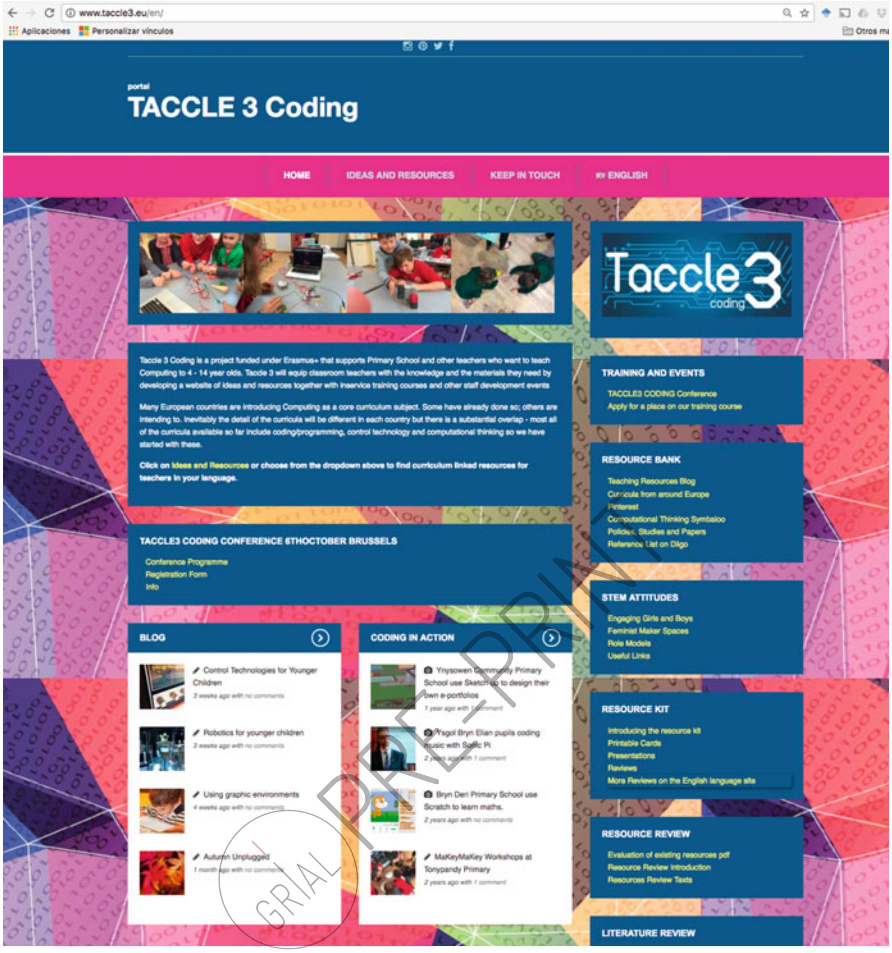
Figure 11.1 shows the main page of the projects website. From this, users may access to different kinds of resources organized by the following categories:

- Using logic
- Algorithms
- Creating + debugging programs
- Controlling things

In Fig. 11.2, the tabs on the top menu correspond to the curriculum areas and underpin the schemes of work that in turn form the basis for the lessons you will be delivering in the classroom. Under each heading, you will find a variety of ideas, lessons, and materials directly related to classroom activities.

One the most interesting TACCLE 3 resources is the activity/lesson. It is published in the form of blog post. Each post explains the basic concept followed

PRE-PRINT GRIAL



GRIAL PRE-PRINT

Fig. 11.1 Main page of the TACCLE 3 website: Source (TACCLE 3 Consortium 2017)

by the aim of the lesson which in turn contributes to one or more of the attainment targets in the computing curriculum.

The outline of the activity follows this scheme:

Title

1. *Overview*

   Brief description
   Age
   Level
   Twenty-first-century skills
   Tips to adapt the lesson (e.g., to older/younger students, students with special needs, etc.)

Fig. 11.2 Categories of the available resources

PRE-PRINT

Material

2. *Aim of the activity*
3. *Needed tools and resources*
4. *Practical activity description*

Figure 11.3 shows an example of a TACCLE 3 activity oriented to introduce the decomposition process, breaking down a problem into smaller manageable parts. Decomposition helps in solving complex problems and managing large projects.

PRE-PRINT
GRIAL

Fig. 11.3 Decomposition activity in TACCLE 3 – Coding project. (Source http://www.taccle3.eu/english/2015/11/16/decomposition/)

3 Developing Smart Textile Objects

To make the abstract learning contents of coding more graspable and usable for teachers at primary school level, the concepts are linked to imagination and phantasies of young children, who can invent and realize their own project ideas to be developed by the learners themselves. The latter is done in project-based learning scenarios (Estruch and Silva 2006; Markham et al. 2003), using embedded sensor- and actuator-based systems, using the Arduino LilyPad technology introduced by Buechley (2014), extended by a visual interface to facilitate programming using icons in a free drag-and-drop environment (Amici) (Kafai 2014).

The technology chosen opens up to link the ideas and imagination to computational thinking (Wing 2006, 2008, 2011) and acting through more art- and craft-based, creative processes. Examples of smart textile projects (and the sketching of electronic circuits) are used. Smart textiles, which are also referred to as "wearables," are a generation of clothes and accessories with embedded microcomputers and offer various possibilities for learning about computational modeling. The system, worn on the body, can respond with behavior programmed by the children themselves. They manipulate and change technology. Using, e.g., conductive yarn (as connector), sensors, motors, and LED lights as well as sewable circuit boards (Arduino LilyPad introduced by Buechley), smart textiles create a link between sensual-haptic materials (Fernández et al. 2016; Scopes and Smith 2010), precise computer control, and creative concepts. New interfaces – sewed, woven, or stitched – can be experienced between body, clothing, and the environment. It can be stitched together with conductive thread to create interactive garments and accessories. In conjunction with the open-source Arduino technology, they open up opportunities for cross-disciplinary teaching of the subjects of art, design, computer science, and music, for example, to address learning in the context of storytelling wearables (Tan 2005), wearable music (Rosales 2012), or sounding artifacts (Trappe 2012).

The Arduino LilyPad technology consists of hard components as well as a programming interface which can be connected to an icon-based interface to be used by younger children at primary school level.

The LilyPad can "sense information about the environment using inputs like light and temperature sensors and can act on the environment with outputs like LED lights, vibrator motors, and speakers" (http://lilypadarduino.org). Kafai (2014) highlighted the LilyPad Arduino kit being a shapable set of technologies, bringing together crafting, design, and technology, supporting individual learning processes.

PRE-PRINT

4 Curriculum Modules for Primary School Teacher Training

The learning activities developed include a teacher training, as well as a tutorial for beginners to programming, which introduces the teacher both to the handling of the LilyPad Arduino hardware and to the application of the Amici user interface and can be used as instructions for teaching processes related to interactive clothing. Also, the development of creative themes is addressed, to support imagination and self-initiated learning. The teacher training is based on the modules identified to develop a project.

The teachers get familiar with the hardware, such as the electronic components, main board, connectors (including unusual wires made of ink or yarn), and sensors and actuators. The teachers use the same modules for project-based learning with physical computing as the school kids in hands-on workshops.

Since, however, the handling of the software and hardware used in the project is documented only insufficiently in Germany, it was decided to write down in a structured way the experiences gained. Although the resultant tutorial does not claim to discuss all software and hardware issues, relevant problems need to be explained in detail. The tutorial was developed on the basis of the EduWear manual compiled by the "Digital media in Education (dimeb)" research group of the University of Bremen (http://goo.gl/a8c2L7).

The following lesson plans for classroom sessions are linked to each other and based on one another. They form the teaching units on developing sensor- and actuator-based systems/developing a project with Arduino LilyPad and Amici software.

Module 1 Getting Familiar with Hardware

This module is part of a series of lesson plans to introduce children (from grade 5 up) to smart textile objects, based and the programming of sensors and actuators set up in an electronic circuit. After the series of lessons 1–6, the learners will be able to develop, connect, and program a sensor- and actuator-based interactive system and contextualize it in a project. Also, there are lessons to introduce the development of electronic circuits through painting connectors (wires) using conductible ink. In those lessons, the learners design and paint electronic systems, which can be integrated in an interactive book project.

a. Aims: familiarizing with the terms and related hardware and understanding the components as a networked system
b. Terms to be introduced: sensor, actuator, connector, main board LilyPad, input, output, and meaning/function in a circuit/interactive system.
c. Methods: relation to sensory perception/the human senses and/or learners to represent the components physically
d. Develop photo work sheets for identification of hardware components, including learning material including exercises

Module 2 Developing an Electronic Circuit

In module 2 of the LilyPad Arduino-based Smart textile introduction series, learners learn to develop a circuit, cable it and make it run by themselves. This way pupils learn: 1. to develop a circuit, cable it and make it run by themselves, 2. how to cable the components using crocodile clips. There are exercises based on work sheets to arrange the components and cables, so that an LED glows continuously or an LED shines on and switches off, come along with the module.

Module 3 Developing an Interactive System: Programming Arduino LilyPad

In this module, the learners learn to program Arduino LilyPad main board by using the icon-based drag-and-drop programming environment Amici. In the session pupils are introduced to Amici software through work sheets with exercises related to LED on/off or for a particular time, in the context of an interactive system. The aim is to make transparent computational thinking and modeling behavior by

PRE-PRINT GRIAL

developing a program. It intends the pupils to understand the computer as a shapable, controllable medium.

Module 4 Programming Arduino LilyPad: Getting Familiar with Amici
The main aim of this module is to make transparent computational thinking/modeling, and algorithms control computer, in order to understand the computer as a shapable, controllable medium to the learners by doing, testing, and debugging. In this module, the issue of testing and debugging is introduced to the pupils by making them develop, test, and debug a program by themselves. The interrelation of such processes which belong together is addressed. Aims are to develop a program, test, and debug it. The issue of bugs and debugging is addressed (original etymological meaning of the bug, esp. for younger kids!). There are exercises to arrange the components and cables, so that a LED glows or so that a LED shines on and switches off.

Module 5 Developing a Project with Arduino LilyPad and Amici
In this module, the learners are encouraged to develop an idea for an interactive project, based on sensors and actuators they know from the previous lessons. By developing an idea for an interactive project, have them identify the tasks to fulfill and the realization by themselves. Co-construction of knowledge is supported and learned through the working and design processes. This learning activity deals with using logic and algorithms.

PRE-PRINT

Module 6 Painting Electronic Circuits
This module deals with particular connectors. Painting electronic circuits can be used as a vehicle to technology education in early age groups. Conductible ink in a pen is used for electronic components in the context of drawing images. As Buechley has stressed, “electronics aren’t just for experts and engineers. Kids and amateurs should be able to play, too.” Buechley (2014) designed paper-based electronics for “sketching” and folding. Teachers like to get and test learning materials which are ready to use in the classroom but also designed flexible enough to be amended individually according to their own purposes, needs, target groups, and ideas. In the following example, learning material is presented. Using conductible ink, the issue of “algorithms” as an endless set of activities which, after its realization, lead to a solution is introduced for primary school level. Therefore, the paper cards (Figs. 11.4 and 11.5) were developed. In Fig. 11.4 the cooking of a pan cake is used as an example for an algorithm.

Fig. 11.4 Drawn algorithm in the form of a game. Learner to put together the images in the right order

Fig. 11.5 If every step is put together correctly, the LED glows

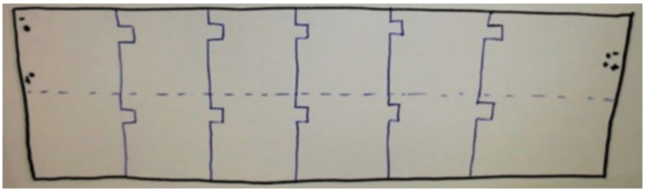


Fig. 11.6 A blanc paper algorithm puzzle is sketched for individual use

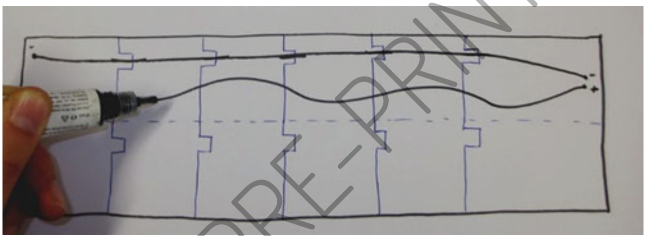


Fig. 11.7 The connection is done using electronic ink

PRE-PRINT
GRIAL

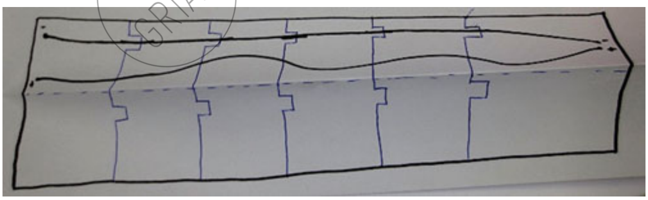


Fig. 11.8 Fold along the dotted line

For initiating the process, a blanc algorithm puzzle is handed out to the pupils (Fig. 11.6).

In step 4, a connection between the ends has to be drawn (Fig. 11.7).

Afterward, it needs folding along the dotted line (Fig. 11.8).

In step 6, pieces are cut apart. Obviously, there is only one correct order of the parts. Here you can see that there will be no electric connection (Fig. 11.9).

In step 7, the parts are folded and numbered. At the front, an algorithm can also be written or drawn (e.g., a recipe) (Fig. 11.10).

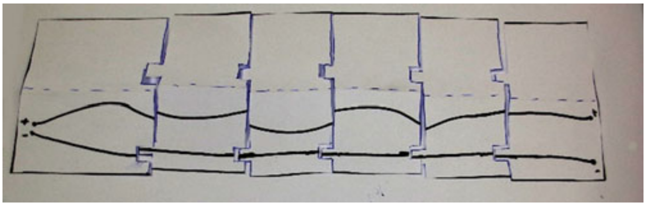
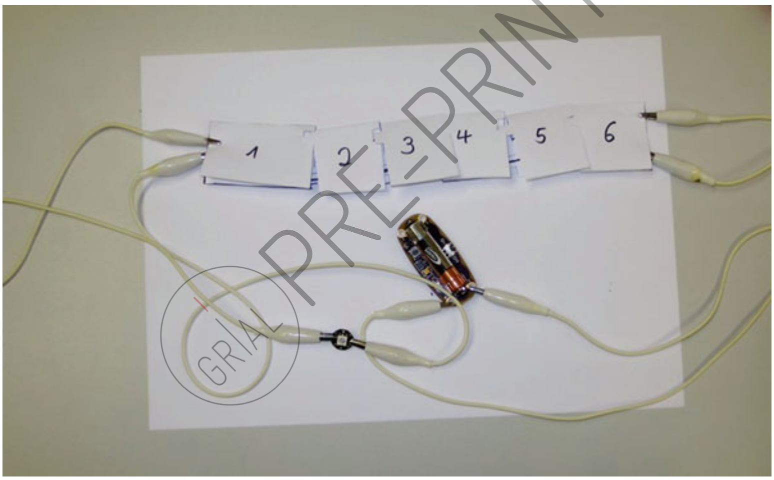


Fig. 11.9 Cutting pieces apart

Fig. 11.10 Fold parts and number them



PRE-PRINT
GRIAL

Fig. 11.11 Cable an actuator and a battery

In the 8th step, an actuator and a battery have to be wired to the end and to the starting point. Then the actuator will react if the algorithm is laid in the correct order (Fig. 11.11).

5 Conclusions

Introducing computational thinking and a solid base of coding is the educational agenda of many countries worldwide. The challenge is to do it in the right way so that the objective is not confused and really influences the acquisition of key twenty-first-century competences; trying to avoid these contents will become in another subject an over-saturated curriculum.

Methods of teaching which have long been overtaken such as the reduction on a tool-oriented and resource-based use of computers, which may now be overtaken, despite all the interdisciplinary and interconnecting efforts, remain as a reality in today's schools, colleges, and outer school contexts.

The presented approaches which were well received by the pedagogical target groups are available for teaching computational modeling at school and university as well as in outside school settings. They can be absorbed and used to ensure a sustainable and systematic integration of computer science contents and embed them into the curricula, crossing the borders of disciplines and school subjects, such as computer science/IT, textile, art, and design education.

In this sense TACCLE 3 project looks for sharing experiences and resources to achieve the pursued goal involving the right actors.

Teachers that are interested in participating in TACCLE 3 – Coding may do it in several ways:

- Visiting the website to access to the resources.
- Writing news related to coding in the schools.
- Making learning activities/lessons.
- Making resource reviews (products, tools, books, courses, etc.) oriented to other teachers. There exists a recommended template (García-Peñalvo 2016e).

Acknowledgments We thank Leah Buechley for sharing the inspiring TED talk and Heidi Schelhowe for making available the software Amici and for testing and modifying eduwear tutorials.

This work has been supported by EU Erasmus+ Programme, KA2 project "TACCLE 3 – Coding" (2015-1-BE02-KA201-012307).

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

PRE-PRINT

References

Aho, A. V. (2012). Computation and computational thinking. *Computer Journal, 55*(7), 832–835. https://doi.org/10.1093/comjnl/bxs074.

Alonso de Castro, M. G. (2014). Educational projects based on mobile learning. *Education in the Knowledge Society, 15*(1), 10–19.

Balanskat, A., & Engelhardt, K. (2015). *Computing our future. Computer programming and coding priorities, school curricula and initiatives across Europe*. Retrieved from Brussels, Belgium: http://fcl.eun.org/documents/10180/14689/Computing+our+future_final.pdf/746e36b1-e1a6-4bf1-8105-ea27c0d2bbe0

Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers and Education, 72*, 145–157. https://doi.org/10.1016/j.compedu.2013.10.020.

Boyd, D. M., & Ellison, N. N. (2008). Social network sites: Definition, History, and scholarship. *Journal of Computer-Mediated Communication, 13*(1), 210–230. https://doi.org/10.1111/j.1083-6101.2007.00393.x.

Buechley, L. (2014). *Crafting the Lilypad Arduino*. Retrieved from http://makezine.com/2014/07/18/leah-buechley-crafting-the-lilypad-arduino/

CEDEFOP. (2015). *EU Skills Panorama (2014) STEM skills Analytical Highlight*. Retrieved from http://skillspanorama.cedefop.europa.eu/sites/default/files/EUSP_AH_STEM_0.pdf

Chajri, M., & Fakir, M. (2014). Application of data mining in e-commerce. *Journal of Information Technology Research, 7*(4), 79–91. https://doi.org/10.4018/jitr.2014100106.

Colomo-Palacios, R., García-Peñalvo, F. J., Stantchev, V., & Misra, S. (2017). Towards a social and context-aware mobile recommendation system for tourism. *Pervasive and Mobile Computing, 38*, 505–515. https://doi.org/10.1016/j.pmcj.2016.03.001.

Estruch, V., & Silva, J. (2006). Aprendizaje basado en proyectos en la carrera de Ingeniería Informática. *Actas de las XII Jornadas de la Enseñanza Universitaria de la Informática (JENUI 2006), Deusto, Bilbao, 12–14 de julio de 2006* (pp. 339–346).

Fernández, C., Esteban, G., Conde-González, M. Á., & García-Peñalvo, F. J. (2016). Improving motivation in a haptic teaching/learning framework. *International Journal of Engineering Education (IJEE), 32*(1B), 553–562.

García-Peñalvo, F. J. (2016a). A brief introduction to TACCLE 3 – Coding European Project. In F. J. García-Peñalvo & J. A. Mendes (Eds.), *2016 International Symposium on Computers in Education (SIIE 16)*. IEEE, USA.

García-Peñalvo, F. J. (2016b). *Presentación del Proyecto TACCLE3 Coding*. Paper presented at the Workshop EI<18. Educación en Informática sub 18, Salamanca, España. http://repositorio.grial.eu/handle/grial/653

García-Peñalvo, F. J. (2016c). *Presentation of the TACCLE3 Coding European Project*. Retrieved from http://repositorio.grial.eu/handle/grial/654

García-Peñalvo, F. J. (2016d). Proyecto TACCLE3 – Coding. In F. J. García-Peñalvo & J. A. Mendes (Eds.), XVIII Simposio Internacional de Informática Educativa*, SIIE 2016* (pp. 187–189). Salamanca, España: Ediciones Universidad de Salamanca.

García-Peñalvo, F. J. (2016e). *Template for TACCLE 3 resources reviewing*. Retrieved from https://doi.org/10.6084/m9.figshare.3545033.v1

García-Peñalvo, F. J. (2016f). What computational thinking is. *Journal of Information Technology Research, 9*(3), v–viii.

García-Peñalvo, F. J., Reimann, D., Tuul, M., Rees, A., & Jormanainen, I. (2016). *An overview of the most relevant literature on coding and computational thinking with emphasis on the relevant issues for teachers*. Belgium: TACCLE 3 Consortium. doi:https://doi.org/10.5281/zenodo.165123.

García-Peñalvo, F. J., Llorens Largo, F., Molero Prieto, X., & Vendrell Vidal, E. (2017). Educación en Informática sub 18 (EI<18). *ReVisión, 10*(2), 13–18.

Guettat, B., Chorfi, H., & Jemni, M. (2010). Automatic update of e-learning environments based on heterogeneous traces. In *Proceedings of the 2nd International Conference on Education Technology and Computer (ICETC 2010)* (pp. V4-512-V514-516). EEUU: IEEE.

Kafai, B. (2014). *Connected code. Why children need to learn programming*. Cambridge, MA: MIT press.

Llorens Largo, F., García-Peñalvo, F. J., Molero Prieto, X., & Vendrell Vidal, E. (2017). La enseñanza de la informática, la programación y el pensamiento computacional en los estudios preuniversitarios. *Education in the Knowledge Society, 18*(2), 7–17. https://doi.org/10.14201/eks2017182717.

Manovich, L. (2013). *Software takes command*. New York: Bloomsbury.

Markham, T., Larmer, J., Education, B. I. F., & Ravitz, J. (2003). *Project based learning handbook: A guide to standards-focused project based learning for middle and high school teachers*. Buck Institute for Education.

PRE-PRINT
GRIAL

National Research Council Committee on Information Technology Literacy. (1999). *Being fluent with information technology*. Washington, DC: National Academy Press.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.

Prensky, M. (2008). *Programming is the new literacy*. Retrieved from http://www.edutopia.org/literacy-computer-programming

Rosales, A. (2012). Wearable music. In *Creating sound effects and music by playing*. Retrieved from Berlin.

Sánchez-Prieto, J. C., Olmos-Migueláñez, S., & García-Peñalvo, F. J. (2013). Mobile Learning: Tendencies and Lines of Research. In F. J. García-Peñalvo (Ed.), *Proceedings of the first international conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'13) (Salamanca, Spain, November 14–15, 2013)* (pp. 473–480). New York: ACM.

Sánchez-Prieto, J. C., Olmos-Migueláñez, S., & García-Peñalvo, F. J. (2014). Understanding mobile learning: Devices, pedagogical implications and research lines. *Education in the Knowledge Society, 15*(1), 20–42.

Sánchez-Prieto, J. C., Olmos-Migueláñez, S., & García-Peñalvo, F. J. (2016a). *A TAM based tool for the assessment of the acceptance of mobile technologies among teachers*. Retrieved from Salamanca, Spain: http://hdl.handle.net/10366/127435

Sánchez-Prieto, J. C., Olmos-Migueláñez, S., & García-Peñalvo, F. J. (2016b, August 22–26). *Technologically reluctant teachers. A TAM based study on compatibility and resistance to change among pre-service teachers*. Paper presented at the ECER 2016, Dublin, Ireland.

Sánchez-Prieto, J. C., Olmos-Migueláñez, S., & García-Peñalvo, F. J. (2017). MLearning and pre-service teachers: An assessment of the behavioral intention using an expanded TAM model. *Computers in Human Behavior, 72*, 644–654. https://doi.org/10.1016/j.chb.2016.09.061.

Scopes, P., & Smith, S. P. (2010). Integrating haptic interaction into an existing virtual environment toolkit. In J. Collomosse & I. Grimstead (Eds.), *Theory and practice of computer graphics*. UK: The Eurographics Association.

Stager, G. S. (2016). Seymour Papert (1928-2016). Father of educational computing. *Nature, 537*, 308–308.

TACCLE 3 Consortium. (2017). *TACCLE 3: Coding Erasmus + Project website*. Retrieved from http://www.taccle3.eu/

Tan, X. L. (2005). *Storytelling wearables*. Retrieved from http://we-make-money-not-art.com/xiao_li_tans_st/

Trappe, C. (2012). Creative access to technology: building sounding artifacts with children *Proceedings of the 11th International conference on interaction design and children, IDC'12 (Bremen, Germany — June 12–15, 2012)* (pp. 188–191). New York: ACM.

Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C., & Verno, A. (2006). *A model curriculum for K-12 computer science: Final report of the ACM K–12 task force curriculum committee* (2nd ed.). New York: ACM.

Wilson, C., Sudol, L. A., Stephenson, C., & Stehlik, M. (2010). *Running on empty: The failure to teach K-12 computer science in the digital age*. New York: Association for Computing Machinery (ACM).

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35. https://doi.org/10.1145/1118178.1118215.

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society a-Mathematical Physical and Engineering Sciences, 366*(1881), 3717–3725. https://doi.org/10.1098/rsta.2008.0118.

Wing, J. M. (2011). Computational thinking. In G. Costagliola, A. Ko, A. Cypher, J. Nichols, C. Scaffidi, C. Kelleher, & B. Myers (Eds.), *2011 I.E. symposium on visual languages and human-centric computing* (pp. 3–3).

Zapata-Ros, M. (2015). Pensamiento computacional: Una nueva alfabetización digital. In *RED, Revista de Educación a distancia,* 46.

PRE-PRINT
GRIAL