Trastea.club, an initiative to develop computational thinking among young students

Mariluz Guenaga
Faculty of Engineering
University of Deusto
48007 Bilbao
Spain
mlguenaga@deusto.es

Iratxe Menchaca
Faculty of Engineering
University of Deusto
48007 Bilbao
Spain
iratxe.mentxaka@deusto.es

Pablo Garaizar
Faculty of Engineering
University of Deusto
48007 Bilbao
Spain
garaizar@deusto.es

Andoni Eguíluz
Faculty of Engineering
University of Deusto
48007 Bilbao
Spain
andoni.eguiluz@deusto.es

ABSTRACT

Trastea.club (http://www.trastea.club/) is an initiative of the University of Deusto (Spain) aimed to develop STEM-related skills among young students (STEM stands for Science, Technology, Engineering, and Science). We chose the name "trastea" because it means to tinker in Spanish, a verb that fits with the hands-on approach of this initiative. Since January 2014, we have organized more than 180 workshops attended by more than 3.200 students from 35 different schools. With the aim of supporting the methodological and technical changes needed to integrate these new skills in the classroom, 216 teachers also benefited from our training courses, consultancy and support to adapt Trastea.club's activities to their curricula. Initiatives such as Trastea.club aim to support the constant adaptation process faced by schools and teachers due to the rapidly changing technologies and the evolution of students' profiles. The main goal of this initiative is to help schools in their constant adaptation processes towards the new digital literacy[1].

CCS CONCEPTS

• Social and professional topics ~ Computational thinking • Social and professional topics~Children • Social and professional topics~Informal education • Social and professional topics~K-12 education

KEYWORDS

Computational thinking; children and K-12 education; technology; STEM.

ACM Reference format:

M. Guenaga, I. Menchaca, P. Garaizar, A. Eguíluz. 2017. *In Proceedings of 5th International Conference on Technological Ecosystems for Enhancing Multiculturality, Cádiz, Spain, October 18 - 20 2017 (TEEM 2017), 6 pages.*
https://doi.org/10.1145/3144826.3145358

1 INTRODUCTION

There is a world-wide concern about STEM education at early ages [1]. The increasing prevalence of computer-mediated activities stressed the need of acquiring basic mathematical and scientific competences from the early childhood, and the benefits of improving academic results in these areas [2-3]. In the last years, the STEM term has evolved to STEAM. The "A" of STEAM stands for Arts and has been included to provide a broader vision of the skills needed for the future, highlighting the transversal nature of STEM and the relevance of creativity and design.

The lack of technology-related workforce foreseen for the coming decades is not the only reason to promote STEAM

[1] *TEEM 2017,* October 18–20, 2017, Cádiz, Spain

ACM ISBN 978-1-4503-5386-1/17/10…$15.00

https://doi.org/10.1145/3144826.3145358

vocations among young students, but can explain the urgency of the latest initiatives in this regard. The estimated rise in overall employment is 3% by 2020, while in STEAM-related professions it is expected to increase by 14% in Europe [4]. Similarly, it is foreseen a rise of 14.8% for all Science and Engineering occupations in the US by 2022.

Figure 1. Make World screenshot.

However, this urgency should not make us forget that we need not only technicians but critical citizens who understand the technology around us and its consequences. We should highlight the human and ethical values of science, and the development of critical citizens to understand the technology around us with a positive attitude [5].

To improve STEAM education's results, children should feel attracted to these subjects at school. This requires engagement and a positive attitude towards STEM [6], and should be done at early ages. Tai found that the aspirations for STEM before 14 yr. is a good indicator of choosing these studies in the future [7]. Therefore, early interventions, maintained over time, favor the choosing of STEM.

Computational thinking is the thought processes involved in solving problems in a way that a computer can execute. Despite its name, Computational Thinking involves a set of skills within STEAM areas not only for computer scientists and engineers, but also for many other professionals. Considering this, an increasing number of countries are integrating it in childhood education [8].

There are many international initiatives to promote computational thinking among students of primary and secondary schools, e.g. the Hour of Code, Scratch Day, or CodeWeek. Also, there is a wide variety of tools [9] and resources available to learn how to code, e.g. Scratch, ScratchJr, Blockly, Alice, Make World and Kodable. These software tools provide block-based programming features to avoid interacting with the complexity of text-based programming languages. For those children, not able to use drag&drop user interfaces, tangible tools and toys have been developed to work on computational thinking skills from a hands-on perspective, e.g. Bee Bot, Kibo, Lego Wedo, Dash and Dot, or Ozobot.

The Faculty of Engineering at the University of Deusto is aligned with the promotion of technological and engineering vocations among pre-university students as part of its communication strategy, but also with the aim of raising social awareness about the relevance of technology to address societal challenges. The Deusto LearningLab (DLL hereafter) research group organizes most of these knowledge transfer activities with teachers and students within the Trastea.club initiative.

Considering the seven powerful ideas of computer science addressed by Umaschi: algorithms, modularization, control structures, representation, design process, debugging, and hardware/software [10], in this paper we will see how the tools used in Trastea.club cover them at different ages.

2 TRASTEA.CLUB FOR STUDENTS

Trastea.club started in January 2014 with the aim of i) increasing the interest of young students on technology, ii) promoting technological and engineering vocations, and iii) researching on how students learn technology and learn with technology. The DLL research group's goal for this initiative is to transfer its knowledge and wide experience in STEM related RDI projects oriented to primary and secondary schools. Resources generated in projects such as Make World, MissToHit or Kodetu are enriched with other existing resources, e.g. Scratch, Arduino, robots, computer science unplugged, etc. to offer a varied range of workshops to 10-14 yr. students.

Student groups are recruited for Trastea.club using the network of the University, as well as the already existing collaborators of DLL. Visits last one morning,where students attend two or three workshops with an average duration of 50 minutes.

Students come to these workshops willing to get a first contact with the university and enjoy carrying out hands-on activities related to technology (programming, robotics, computing). Considering the feedback collected from their post-visit surveys, Trastea.club fulfills their expectations.

2.1 Tools and resources

As mentioned before, Trastea.club's workshops use both own and third-party developments. Among the tools developed by ourselves we have Make World (see Fig. 1, https://makeworld.eu), an online platform that merges the learning of STEM concepts with programming, challenges that have to be solved through block-programming [11]; Kodetu (http://kodetu.org/), a set of computational thinking challenges of increasing difficulty (see Fig. 2) with research-oriented features (i.e. interaction data collection for analysis, gamification, generalization, etc.); Unplugged computing, a series of games that introduce computer related concepts without technology (i.e. poker cards [12] and Compus (see Fig. 3, http://compus.es/), a board game where the player is the CPU).

We also use widely-known resources to spread the passion for technology, such as Makey Makey, Scratch, Arduino, AppInventor, Lego Wedo, Bee bot, and BQ robot.

2.1.1 Kodetu.org. Kodetu is one of the tools developed by LearningLab. It is an online platform for learning the basics of programming. Based on "Maze", it is a basic application developed with Blockly (a software library by Google to create visual programming environments). We have modified it in three ways: 1) we added five new challenges to support participants understand block-based coding; 2) we added a fine-grained logging system to collect users' actions while using the tool; and 3) we developed a system for defining different settings for groups of users.

To solve Kodetu's challenges students have to define the astronaut's movements using coding blocks in a workspace. There are blocks to go forward (F), turn right (R), turn left (L), check whether there is a path in front, left, or right of the astronaut, and perform loops to move toward the goal.

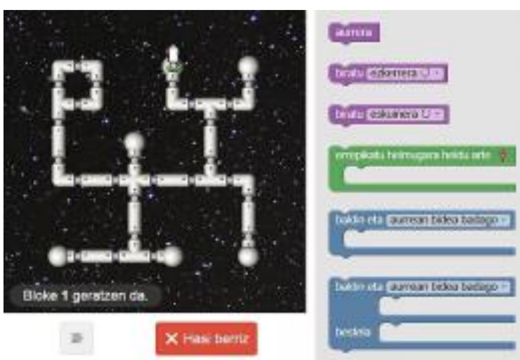

Figure 2. Kodetu screenshot

2.1.2 Compus is a board game COMPUS is a game for 1 to 4 people with an estimated minimum age to play of 8 years and older. In COMPUS each player will simulate being a computer program and he/she will try to get the desired result (a combination of bits) before the others.

The goal of the game is to get the desired result in the register A of the CPU. The first player to achieve this goal wins the game. Every turn, each player must do the following: 1) know how much CPU time is available, 2) play as many operation cards as time units have been assigned in the previous step, and 3) take as many operation cards as they have been played in the previous step. Operation cards include logic and arithmetic operations, sucha as: increment, decrement, or, and, not, xor, movement, rotate left and rotate right.

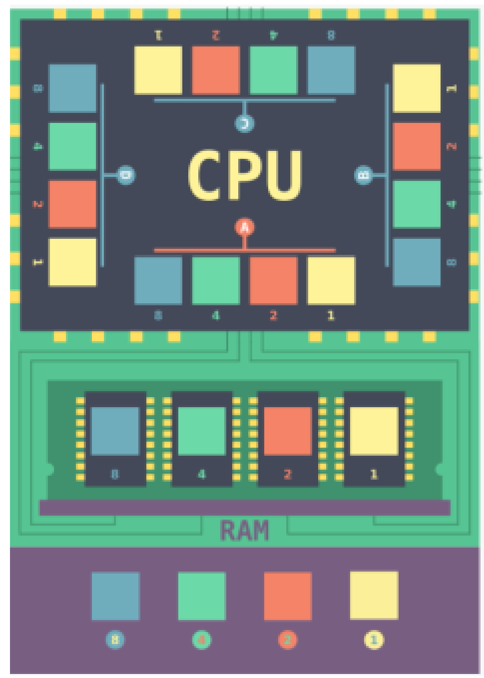

Figure 3. Compus board game.

3 TRASTEA.CLUB FOR TEACHERS

As result of Trastea.club workshops with students, most teachers involved showed interest in integrating computational thinking skills in their classes. To respond to this demand of training and support, we created Trastea.Club for teachers.

The main goal of Trastea.club-teachers was to provide a place to support teachers in the integration of programming, robotics and related issues in their teaching. In January 2015, we had the first meeting with them, and more than 200 teachers have benefited from this initiative since then.

The keys for the collaboration are the following: 1) DLL offers consultancy, training and support on what, how and when they could integrate these tools and methodologies in the classroom; 2) every 1.5-2 months the DLL research group's facilitators arranges a meeting to address the needs identified either by them or by teachers; and 3) a brief training is followed by discussion and reflection about methodologies and tools, the sharing of good practices and their implementation in the classroom.

Trastea.club-teachers is open and free, with a maximum of 40 teachers in each session. In return, teachers acquire the commitment to collaborate in group's research through their feedback on projects, testing and piloting, dissemination of outcomes and support as real target end-users of DLL's activity.

Most participant teachers need advice on which technology they could use in each level, and how to use it in class. Based on our experience, we propose integrating different tools at each educational level (see Table 1 to 3), considering also their price, so there is always a free or low-price option. We recommend each school to design a roadmap that integrates a varied set of tools at different levels. Their use should not be tight to computer science or technology subjects, but in any subject or in multi-subject projects.

Table 1: Programming tools for each level

Tool	Primary school			Secondary school	
	1-2	3-4	5-6	1-2	3-4
Kodetu	X	X	X	X	
Scratch		X	X	X	XX
Make World	X	X	X	X	
App Inventor				X	X

Table 2: Robotics tools for each level

Tool	Primary school			Secondary school	
	1-2	3-4	5-6	1-2	3-4
Bee bot	X	X			
Lego Wedo		X	X	X	
Arduino				X	X

Table 3: Digital culture tools and resources for each level

Tool	Primary school			Secondary school	
	1-2	3-4	5-6	1-2	3-4
Unplugged		X	X	X	X
Compus		X	X	X	X
Gender&Tech	X	X	X	X	X
SocialLab			X	X	X
Creative electronics	X	X	X	X	
Digital identity			X	X	X

3.1 Training and supporting

Training was demanded by teachers from the very beginning. Despite the wide training offer in the market at a reginal level organized by the Basque Government, we realized that Trastea.club-teachers cover an unattended demand: instead of teaching how to use the tool or technology, our training courses focus on the proper integration of these tools in the curriculum, with a special effort in porting the experiences and methodologies to the subject of each teacher.

Considering the variety of areas, we designed short training modules of Scratch, Arduino or robotics where teachers had to develop an activity for their classroom with the support of DLL's experts.

Longer training modules followed a blended model, with five hours face-to-face and 20 hours of online activities tutored by DLL team. As a result of the training, participants have to develop a teaching session for their subject and its corresponding lesson plan. It is worth to highlight that activities and lesson plans developed by teachers for their subjects are shared for free at the website, so the educational community can benefit.

We also advise management teams and schools' coordinators on how to carry out global integration of computational thinking at the center. These strategies are usually planned in a framework of two or three years and there are already several centers implementing these models

4 TRASTEA.CLUB AND THE SEVEN POWERFUL IDEAS OF COMPUTER SCIENCE

Algorithm

Any of the programming tools used in Trastea.club require a set of finite and concrete steps that lead to the correct solution. In these cases, students are developing the so-called algorithm. When they follow the instructions to build something, i.e. a Lego structure or a science experiment, they are executing an algorithm programmed by the author of the activity.

Modularization

When students decompose a complex problem into smaller and easier sub-problems, they are following a top-down approach to solve it. The independent solutions of these sub-problems and their integration involve modularization. Each sub-solution is a module that combines with the rest to solve the global problem. This is design modularization.

When coding, modularization is implemented through procedures in Make World and messages in Scratch. In both cases, a set of instructions is encapsulated to provide a solution to part of the problem, better organize the code, debug more effectively and reuse these modules.

Control structures

Sequences are used in all the tools we use for programming in Trastea.club, e.g. Bee bot only has this possibility. Simple loops (play forever) are incorporated in Make World, and more complex structures (simple and double conditionals, other type of loops) are presented in Kodetu, AppInventor and Scratch.

Representation
Scratch and AppInventor support different ways to abstract and represent data used with algorithms to obtain specific goals such as walking a given distance, asking user for information, etc.

Design process
Students should *Ask* which is the objective of the challenge they face (e.g. get the astronaut to the goal in Kodetu, complete the water cycle in Make World, or create an interactive story in Scratch). Then, they have to *Imagine* the solution and *Plan* the steps to achieve the goal. After that, they *Create* the program or structure, *Test it* and *Improve it* if necessary. Finally, they *Share* the solution with peers, teachers or the community.

Debugging
Try&error is one of the mainly used strategies among young children. They are not afraid of making mistakes: when their solution does not work, they review the behavior of the program, their solution and contrast what should have happened with what it actually happened. In some cases, i.e. Scratch, Kodetu and Make World, they have visual clues about the program execution so they can figure out the part of the solution that does not behave as expected.

Hardware/software
Both hardware and software tools are used in the workshops. Offline and online applications-software- accessed through computers, mouse and keyboard-hardware-, but also robots, building blocks and cards.

5 RESULTS

As you can see in the previous section, the seven powerful ideas of computer science described by Umaschi are widely covered by different tools and workshops organized by Trastea.club. Integrated in each proposal students acquire knowledge and skills useful not only for programming, but to address many other challenges beyond computational thinking.

Table 4 shows the number of workshops organized by DLL per academic year, as well as the number of participants. In addition, we have organized events aimed at the same audience, such as Scratch Day-Bilbao, Edutinker international conference (2016 and 2017), Digital Education workshop that reached the ninth edition and ForoTech, the engineering fair at the University of Deusto. These events also attracted a high number of participants.

Table 4: Trastea.club workshop participants

Academic year	Workshops	Students	Schools	Teachers
2014-15	55	1,450	10	50
2015-16	73	2,063	18	110
2016-17	107	2,675	32	130

Figure 4 shows the distribution of workshops developed from 2015 to 2017 within Trastea.club.

Figure 4. Distribution of workshops 2015-2017.

6 CONCLUSIONS

Contrary to vendors' beliefs, the overwhelming offer of educational tools to develop computational thinking skills discourages most teachers from incorporating this type of activities into their classes. Teachers tend to see most of these technologies as complex and unrelated to their curricula. Thus, most students lack fundamental skills for the new digital literacy. With the aim to cover this gap, the Faculty of Engineering of the University of Deusto launched Trastea.club, focused both on students and teachers.

The feedback in both cases has been very positive. Students, both boys and girls, enjoy the sessions and declare to learn how to use interesting and challenging tools that allow them to develop games, control robots or understand computer science related concepts. They are not mere users, but creators of applications using tools such as Scratch, Arduino or Make World. As shown above, the concepts developed in Trastea.club's workshops are aligned with the seven powerful ideas from computer science described by Umaschi.

In the case of teachers that joined Trastea.club-teachers, they showed a great interest in training. They specially valued the pedagogical approach of the coursed, not mainly oriented to learn the functionality of the tools, but to the curricular integration in their subjects. After a 25 hours blended course, they are happy to come out with an activity ready-to-use in their classroom.

Once the most motivated and innovative teachers have joined Trastea.club and completed our courses, the challenge is to attract those that are not so convinced about this kind of approaches. However, many schools are adopting an integral plan of including technology and computing in their curriculum from the early stages, so many teachers will need the training and support offered by Trastea.club-teachers.

In the near future, computer science will be no more an independent subject, but a transversal skill addressed in many subjects. In this regard, new vcrsions of official curricula are already addressing technological education. For example, algorithms and programming languages have been recently added as concepts to be learnt in primary and secondary schools of the UK and other countries.

Among the challenges Trastea.club faces, we consider the continuous development of new tools and resources to develop computational thinking skills, both tangible, unplugged and software; the evolution in student's profile, closer to these areas; and the integration of computing in schools. This will require a continuous update and adaptation to the audience needs.

We planned to introduce two new courses in the academic year 2017-2018. The first one is about the use of mobile devices, telephones, in STEM education. We want teachers to make the most of such a powerful device, using their sensors, apps and other features. Instead of forbidding the use of telephones in the classroom, we want to encourage their educational use. The second novelty is a course about Evidence Based Education. In a world where so many trends and innovations emerge every day, educators need clear guidance on which of them are proven to be effective and in which circumstances. Also, they will learn how to measure the impact or their innovations in the classroom.

DLL will follow the activity in R&D&I projects (Erasmus+, H2020) and transferring the knowledge generated to this kind of initiatives. We aim at working with the educational community so both benefit from the exchange of knowledge and experience.

REFERENCES

[1] 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: TACCLE3 Consortium. doi:10.5281/zenodo.165123.

[2] Anderson, L.W., Krathwohl, D.R., Airasian, P.W., Cruikshank, K.A., Mayer, R.E., Pintrich, P.R., Raths, J., Wittrock, M.C. 2001. A Taxonomy for Learning, Teaching, and Assessing: A revision of Bloom's Taxonomy of Educational Objectives. New York: Pearson, Allyn & Bacon.

[3] Bloom, B.S. (Ed.). Engelhart, M.D., Furst, E.J., Hill, W.H., Krathwohl, D.R. 1956. Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain. New York: David McKay Co Inc.

[4] Balanskat, A., & Engelhardt, K. 2014. Computing Our Future: Computer Programming and Coding-Priorities, School Curricula and Initiatives Across Europe.

[5] Gago, M. 2014. How Ministries of Education should uptake STEM challenges? 2nd Scientix Conference, 24-26 October 2014, Brussels.

[6] Kier, M. W., Blanchard, M. R., Osborne, J. W., and Albert, J. L. 2014. The Development of the Stem Career Interest Survey (StemCis). Research in Science Education, (44), pp. 461481.

[7] Tai, R. H., Liu, C. Q., Maltese, A. V., & Fan, X. 2006. Planning early for careers in science. Life sci, 1, 0-2.

[8] Marina Bers, L. Flannery, E. Kazakoff, and A. Sullivan. Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. Computers & Education, vol. 72, pp. 145-157, 2014.

[9] Hughes, J., 2016. Best apps for teaching programming. TACCLE 3, http://www.taccle3.eu/english/2016/05/10/best-apps-for-teaching-programming/

[10] Marina Bers. Coding as a Playground Programming and Computational Thinking in the Early Childhood Classroom. Taylor and Francis Group (in press).

[11] Mariluz Guenaga, Iratxe Mentxaka, Pablo Garaizar, Andoni Eguiluz, S. Villagrasa, I. Navarro. 2017. Make World, A Collaborative Platform to Develop Computational Thinking and STEAM. In: Zaphiris P., Ioannou A. (eds) Learning and Collaboration Technologies. Technology in Education. LCT 2017. Lecture Notes in Computer Science, vol 10296. Springer, Cham

[12] Pablo Garaizar. 10 experimentos con cartas: Informática desenchufada (Ciencia Infinita). Ed. A Fortiori. At: http://www.cienciainfinita.com/2016/06/10-experimentos-con-cartas-informatica-desenchufada/ Last access July 2017.