Education 4.0 and computer science: A European perspective

Author 1, Author 2
Author's Common Affiliation
Department/Institute
Full Address
author1, author2@domain.com

Author(s) Name(s)
Author Affiliation(s)
Department/Institute
Full Address(es)
E-mail(s)

Abstract. *This systematic literature review aimed at identifying the pedagogical approaches, aligned with Education 4.0, used to support teaching computer science courses with undergraduate and graduate students in Europe. A three-step coding process was conducted to identify and analyse 20 papers. Quantitative and qualitative analysis of the selected papers revealed a 3-cluster solution with common characteristics that could be used to describe those pedagogical approaches. The review also showed that the term Education 4.0 is still relatively new and has not been conceptualized in terms of computer science courses, although the characteristics of Education 4.0 are visible throughout the pedagogical approaches.*

Keywords. Computer Science, Education 4.0, Systematic Literature Review.

1 Introduction

Driven by Industry 4.0 and digital technology, jobs are becoming more flexible and complex. People's capacities to be entrepreneurial, manage complex information, think autonomously and creatively, use resources, including digital ones, smartly, communicate effectively and being resilient are more crucial than ever. Therefore, Computer Science (CS) students need to be equipped with a different set of skills than before that would enable them to learn how to learn anytime anywhere, to become independent learners, to be exposed to more project-based and hands-on learning (Author A, 2021b; Garousi, Giray, Tüzün, Catal, & Felderer, 2019; Hussin, 2018).

As evidenced by a range of studies (Aničić, Divjak, & Arbanas, 2017; Llorens, Berbegal-Mirabent, & Llinàs-Audet, 2017; Wasson & Kirschner, 2020), while substantial progress has been made over the years in nurturing CS graduates in Europe in comparison to the US and Asian countries, some argue that European CS programmes lack innovation and focus on (softer) skills. In line with Industry 4.0, and building on a range of conceptualisations of Education 4.0 (Fisk, 2017; Hussin, 2018; Salmon, 2019), we define Education 4.0 as an "approach to learning and teaching that emphasises the development of skills and competences necessary in a modern workplace using up-to-date technology. The skills and competences developed may relate directly to the technology, or they may be the softer skills (such as team-working and creativity) that are needed to work effectively in such an environment. The approach involves the use of technology and/or pedagogy that is innovative in the context, and therefore requires flexible and creative approaches to its implementation" (Author A, 2021a).

In this study we aim to conduct a systematic literature review (SLR) on Education 4.0 literature published on innovative CS practice in Europe. SLRs are useful to identify, evaluate, and summarise the findings of all relevant individual studies over particular area of research (Hattie & Yates, 2013; Mangaroska & Giannakos, 2019), in this case with the objective to identify skills and competencies that were highlighted by authors when describing innovative practices.

2 Computer science and Education 4.0

A range of SLRs on CS and innovative approaches to teaching and learning have been published in the last five years (e.g., Aničić et al., 2017; Garousi et al., 2019; Scatalon, Garcia, & Barbosa, 2020). For example, Aničić et al. (2017, p. 192) conducted a meta-analysis of 155 papers from 1980-2014, with the main aim of "give insight into the current research on the education and career development of graduates in the field of ICT". A broad range of search terms were used, whereby the findings in terms of curriculum design and delivery indicate a need to adjust the curricula to the needs of industry. As argued by Aničić et al. (2017, p. 194) "the literature indicates a need for innovative approaches in curriculum design and delivery, such as designing competency-based programs that are not restrained by the traditional semester seat-time model, providing flexible curriculum and minimizing the time spent in the classroom, or offering courses on not only how to manage innovation, but also on how to innovate". In terms of teaching methods that could help to encourage graduate employability include learning by doing, learning from mistakes, team-work and collaborative learning. Furthermore, a job-oriented experiment course system, problem- or project-based learning and work-integrated learning to develop a wide range of desired skills, competences and knowledge are encouraged, which link well with some of the Education 4.0 concepts.

Garousi et al. (2019) specifically focussed on how software engineering education was aligned with industrial needs. Using a SLR of 34 papers in period 1995-2018 they identified eight research questions, whereby two are highly relevant to our project (What curriculum models (bodies of knowledge) have been used to design the studies?; What educational recommendations are provided in each study?). Garousi et al. (2019, p. 77) indicated that the "qualitative coding provided four themes for the educational recommendations provided in the papers: (1) Need for more emphasis on soft skills (20 papers), (2) Need for active Infrastructure as Code (IAC) (3 papers), (3) Less emphasis on certain topics (2 papers), and (4) Other recommendations (7 papers)." In order to encourage development of soft-skills, Garousi et al. (2019) encouraged educators to use real-life projects, implement industry-academia collaboration in the design of education, and to anticipate future trends, while also preparing students to deal with them.

Based upon 195 empirical papers Scatalon et al. (2020) provided an overview of the practices that have been used to integrate software testing into programming education. The study showed that testing practices in programming assignments involved students to different extents: analysing test results from submission tools, working with instructor-provided tests, using support mechanisms to design tests (e.g. plugins where students insert inputs and expected outputs) and, finally, students writing their own tests. However, very few studies addressed how students learned testing concepts in programming courses.

While these SLRs provide important and deep insights into how CS, computer programming, and AI have been used in a range of HE contexts, none of these studies specifically focus, mention, or include concepts of Education 4.0. Definitions vary but usually focus on innovation, novelty, use of technology, and connections with employment and industry (Hussin, 2018; Jisc, 2019; Salmon, 2019). The number 4.0 makes a connection with the view that there have been four industrial revolutions with the current Industry 4.0 increasingly automated, making use of modern smart technologies and the Internet of Things (objects that can exchange information over the Internet). Fisk (2017) and later on Hussin (2018) identified nine trends associated with Education 4.0:

1 Learning any time / anywhere: Students will be able to learn where and when they choose.

2 Personalised learning: Study tools will adapt to the capabilities of the student.

3 Choice how to learn: Students will be able to modify their learning process.

4 Project-based learning: Students will learn to apply their skills in a variety of situations.

5 Hands-on learning: Students will have authentic experiences and gain real-world skills.

6 Data interpretation: Students will learn to interpret and reason with data.

7 Assessed differently: Knowledge and skills will be assessed in new ways.

8 Student ownership of curriculum: Students will have critical input into their courses.

9 More independent: students will become more independent.

In particular with the COVID-19 Pandemic and the rapid shift to online education, it is essential to update our insights about how CS teachers are adopting innovative pedagogies and Education 4.0 approaches in Europe. In order to investigate which innovations are being introduced in the field of CS in Europe, a SLR was carried out, focusing on two research questions.

RQ1: Which pedagogic approaches are used to support the teaching of computer science to undergraduate and postgraduate students in Europe?

RQ2: Which of these approaches align with Education 4.0?

3 Methods

Four research databases were searched: Science Direct, Wiley InterScience, Web of Science, and Scopus. These were chosen because of their ranking as academic research databases, and good coverage of relevant studies relevant for the review.

3.1 Initial inclusion and exclusion criteria

Papers had to be published in English during the five-year period 2016–2020, keywords had to include Computer Science; undergraduate and/or postgraduate; as well as education, teaching and/or pedagogy. The following search string was used: "computer science" AND education AND teaching AND pedagogy AND ("undergraduate" OR "postgraduate"). These search terms identified 231 unique publications across the four databases. Publications identified using the search criteria were excluded if any of the following exclusion criteria applied: 1) The focus is on primary and/or secondary education; 2) The focus is on a subject other than Computer Science; 3) The focus is on learners (e.g., their gender or expectations) rather than teaching; 4) The study was not conducted in Europe (incl. UK).

3.2 Coding process

In Phase 0, Author RF manually screened the abstracts to check whether the respective papers should be included or excluded based upon the above criteria. Subsequently, 75 papers were excluded. In Phase 1, following a one hour online training and discussion of the online coding scheme of four variables, 156 papers were read in-depth by 18 members of project [blinded for peer review] and based upon two inclusion criteria (i.e., 1. is it an "innovative" application in a CS course; 2. Does it use technology or pedagogy in an innovative way). By including experts from CS and educational technology from six EU countries we aimed to develop an inclusive multi-disciplinary team of coders to analyse the literature. On average the members coded 8.26 papers (range: 3-11), whereby 68 papers were included for subsequent analysis. All papers were annotated and uploaded in Google drive for a second round of coding.

In Phase 2, 17 members of project [blinded for peer review] participated in a follow-up one hour online training and discussion of the online coding scheme of 20 variables. Coders were randomly allocated a new set of papers to code in comparison to their initial coding in Phase 1. The 17 members coded on average 4.25 papers (range: 2-10) based upon the coding scheme developed from the above research questions. Afterwards, the first coders from Phase 1 checked the codings from the second coders in Phase 2, discussed any differences (i.e., 17 times (1%)), and agreed on the final coding. A random sample of 15 papers was double coded and indicated reliable coding. Removing any paper which did not meet our definition of Education 4.0 or received a 0 score on Hussin (2018), we ended up with 66 papers. For the analysis we used both the individual scores as well as the aggregate score of Hussin (2018). In total 20 (27%) of papers were written about CS practices in Europe and subsequently analysed for this study.

4 Results

As indicated in Figure 1, CS studies identified were mainly from Spain (6), Germany (4), Finland (2), Greece (2), France (1), Ireland (1), Norway (1) as well as from Sweden (1) and the UK (2). Obviously this does not mean that in other EU countries no research on Education 4.0 in CS is conducted, and the findings might be different if other search strings are used.

The vast majority of studies included referred to undergraduate CS students (66%) (e.g., Apiola, Lokkila, & Laakso, 2019; Knobelsdorf, Frede, Böhne, & Kreitz, 2017), followed by a mix of undergraduate and post-graduate students (Goumopoulos, Nicopolitidis, Gavalas, & Kameas, 2017; Urquiza-Fuentes, 2020). Two studies did not explicitly mention the student population under study (Llorens et al., 2017; Schäfer, 2019).

In terms of RQ1 and RQ2, perhaps surprisingly none of the 20 articles explicitly mentioned "Education 4.0". In part this could be a result of the relatively recent conceptualisation of Education 4.0, and in part this could be due to the lack of adoption of the term Education 4.0 in the specific discipline of CS. Based upon our broad definition of Education 4.0 in total 14 articles (78%) were coded to fit under this definition. Furthermore, 20 articles included at least one Hussin (2018) Education 4.0 characteristic.

Figure 1 Location of studies in Europe

As indicated in Figure 2, on average the 20 articles included 4.10 out of nine Education 4.0 characteristics of Hussin (2018), with a substantial variation (SD = 2.10). There seemed to be two peaks in Figure 2, whereby seven studies only had two Education 4.0 characteristics, while another peak around 4-5 Education 4.0 characteristics was present.

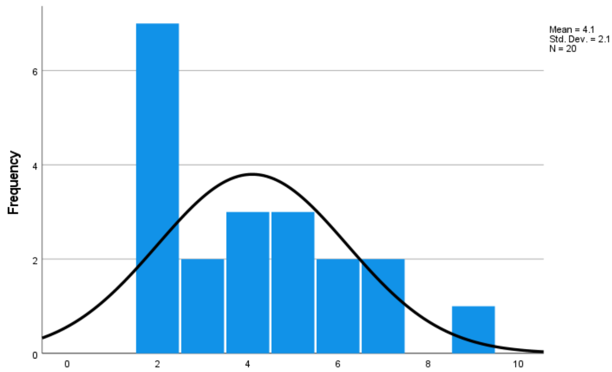

Figure 2 Histogram of Education 4.0 (Hussin, 2018)

Amongst these 20 studies, the most common Education 4.0 characteristic was "9) students will become more independent in their own learning" (75%), followed by "5) students will be exposed to more hands-on learning through field experience" (70%), "1) learning can be taken place anytime anywhere" and "4) students will be exposed to more project-based learning" (both 55%). Less than half of the studies included "2) learning will be personalized to individual students (40%), "3) students have a choice in determining how they want to learn" (35%), "7) students will be assessed differently and the conventional platforms to assess students may become irrelevant or insufficient" (30%), and finally "6) students will be exposed to data interpretation" and "8) students' opinion will be considered in designing and updating the curriculum" (each 25%).

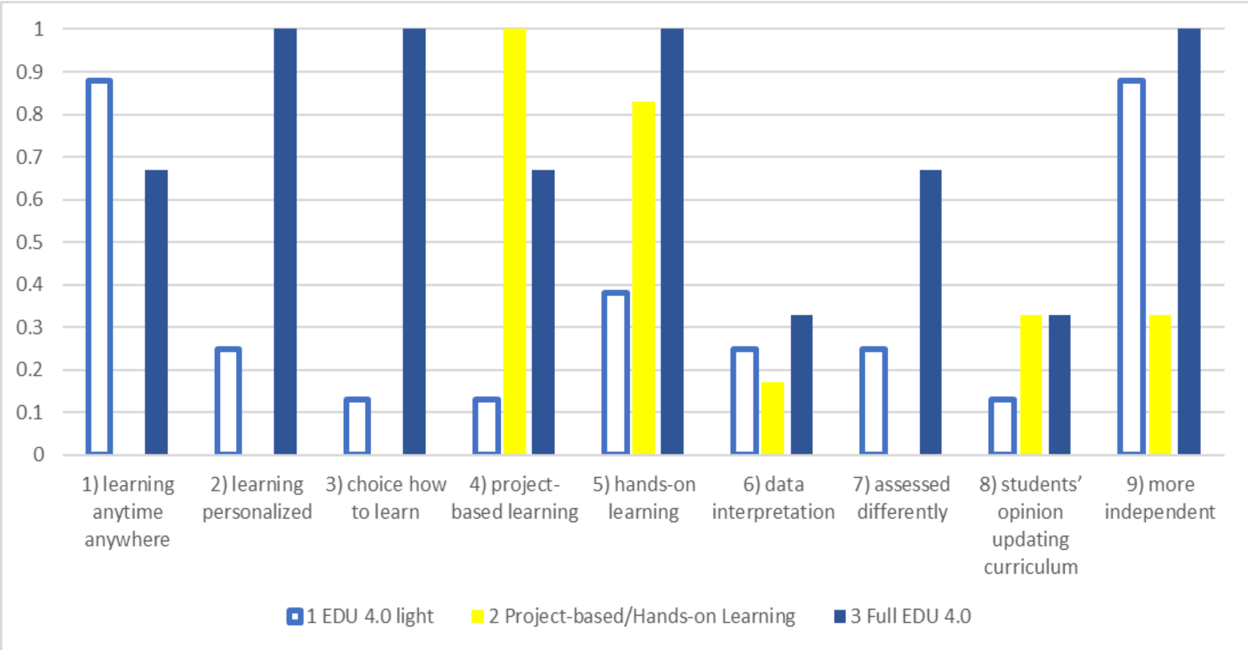

Figure 3 Cluster analysis of EDU 4.0 (3 cluster solution)

Finally, a follow-up analysis using K-means cluster techniques indicated a three-cluster solution across the 20 papers. As illustrated in Figure 3, there seemed to be 3 clusters of papers, which we label as 1) *EDU 4.0 light* (n = 8), 2) *project-based/hands-on learning* (n = 6), and 3) *full EDU 4.0* (n = 6). With the notable exceptions of Hussin "6) students will be exposed to data interpretation" and 8) "students opinion will be considered in designing and updating the curriculum", using ANOVAs all the Hussin characteristics were significantly different between the three clusters with large effect sizes. One potential reason why we did not find significant differences between the three clusters on these two Hussin characteristics was the relatively low use in the 20 studies. In other words, there appeared to be three distinct innovative pedagogical practices present in Europe in published work on CS in the last four years.

As indicated in Figure 3 EDU 4.0 light studies (blue and white bar) mostly had relatively low Hussin (2018) total scores, and mainly focussed on learning anytime anywhere and allowing students to become more independent. In contrast, project-based/hands-on learning studies (yellow) mostly focussed on project-based and hands-on learning, with no room for anytime/anywhere, personalised or choice how to learn. Finally, the full Edu 4.0 cluster studies (blue bar) mostly used the full range of options, with the exception of data interpretation and including student voice in updating curriculum. In the remainder of this study we will describe each cluster in particular.

Table 1 European Perspectives on EDU 4.0

Authors	H 1	H 2	H 3	H 4	H 5	H 6	H 7	H 8	H 9	EU country
Edu 4.0 Light										
Aghaee and Keller (2016)	Y			Y	Y		Y		Y	Sweden
Apiola et al. (2019)	Y		Y		Y				Y	Finland
Degener, Haak, Gold-Veerkamp, and Abke (2019)	Y								Y	Germany
Dondio and Shaheen (2019)								Y	Y	Ireland
Parejo et al. (2020)	Y						Y			Spain
Knobelsdorf et al. (2017)	Y	Y			Y	Y			Y	Germany
Schäfer (2019)	Y								Y	Germany
Urquiza-Fuentes (2020)	Y	Y				Y			Y	Spain
Project-Based										
Carrascal, del Barrio, and Botella (2021)				Y	Y	Y		Y		Spain
Casañ, Alier, and Llorens (2020)				Y	Y					Spain
Cobos and Roger (2019)				Y				Y		Spain
Fagerholm et al. (2018)				Y	Y				Y	Finland
Llorens et al. (2017)				Y	Y					Spain
Mäkiö, Yablochnikov, Colombo, Mäkiö, and Harrison (2020)				Y	Y				Y	UK
Full EDU 4.0										
Broisin, Venant, and Vidal (2017)	Y	Y	Y		Y				Y	France
Charlton and Avramides (2016)		Y	Y	Y	Y		Y		Y	UK
Goumopoulos et al. (2017)	Y	Y	Y	Y	Y	Y			Y	Greece
Pawelczak (2017)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Germany
Troussas, Krouska, and Sgouropoulou (2020)	Y	Y	Y		Y		Y	Y	Y	Greece

4.1 EDU 4.0 light

In the eight EDU 4.0 light studies, teachers mostly focussed on more independent learning and learning anytime anywhere (each 88%). Furthermore, there is some focus on hands-on learning (38%). For example, Schäfer (2019) introduced the concept of a modern C ++ course for students of CS and electrical engineering based on an inverted classroom and with pleasant Internet of Things (IoT) hardware. The main goal of this new course was to reduce lecture time in favour of practical learning of students through programming. Schäfer (2019) used an inverted classroom to adapt the pace of teaching to the individual needs of students to enable them to study anytime and anywhere. In his conception of the course, Schäfer (2019) recommended replacing theoretical lectures with discussion meetings between teachers and students, and most of the time defined by students' practical work on a programming project.

Aghaee and Keller (2016) monitored how an ICT-based support system facilitated peer interaction (i.e., peer reviews, active participation, and final opposition) in thesis production at undergraduate and postgraduate levels in Sweden. This process enabled both authors and reviewers to learn and to improve their theses. The learners perceived the peer interaction useful to enhance the quality of the thesis outcomes.

In a study in Germany, Degener et al. (2019) integrated LEGO MindStorms EV3 robots within lessons teaching the programming language ANSI-C. The intention was to make practical programming lessons more tangible and closer to the future field of work for CS and engineering students. Although the EV3 contributed to learning success and fun during the lessons, students were not able to program the EV3 outside laboratory opening hours and practical lessons. To solve this problem, a simulation was provided to make the programming task time-and location-independent.

In a study in Spain, Parejo et al. (2020) flipped a course on software architecture and integration, that formed part of a Software Engineering degree. In addition, the gamified platform Kahoot was used for interactive tests at the beginning of the laboratory sessions. Based on the answers to these quick quizzes, the lab instructor decided which concepts to clarify. Students had, on average, 24 more minutes per session to solve in-class exercises when using the flipped-classroom approach. More than 70% of students considered the quantity, duration and didactic content of the videos (very) appropriate; and 90% of students preferred this approach for laboratory sessions. While each of these studies indicate substantial innovative pedagogical enhancements, most of these studies focussed only on some of the Education 4.0 characteristics.

4.2 Project-based/hands-on learning

The second cluster of six studies had a strong focus on Hussin (2018) project-based (100%) and hands-on learning (83%). For example, Casañ et al. (2020) provided a critical review of 29 years of teaching courses on social, environmental, and ethical issues to students of Informatics Engineering in Spain. Strategies include case study sessions and active methodologies. Collaborative approaches include the jigsaw method, think-pair-share, group investigation, and role-playing debates in online forums. Over time, the use of wikis to support collaboration has given way to use of Google Drive.

In Finland, Fagerholm et al. (2018) implemented a course where students were considered as prospective entrepreneurs, as well as potential employees in modern, start-up-like intrapreneurship environments within established companies. This paper reported on experiences gained during seven years of teaching start-up knowledge and skills, whereby a Software Factory, an educational environment for experiential, project-based learning, was developed.

In a UK context, Mäkiö et al. (2020) implemented a Java programming course using a task-centric holistic agile teaching approach (T-CHAT) to enhance both technical skills and transferable skills of students. T-CHAT integrated five pedagogical approaches: 1) perceptional teaching (moving from observations to explanatory models), 2) project-based learning, 3) problem-based learning, 4) research-based learning, and 5) face-to-face teaching. In all six studies there was a strong focus on hands-on and project-based learning, allowing CS graduate to develop strong programming and soft-skills, often working in team. However, due to the nature of project-based learning there was relatively low flexibility in terms of anytime/anywhere, personalisation, and choice of study.

4.3 Full EDU 4.0

The third and final cluster which we labelled as the full EDU 4.0 version was strongly focussed on personalised learning, choice how to learn, hands-on learning becoming more independent (all 100%), learning anytime anywhere, project-based learning and assessed differently (each 67%). In a French context Broisin et al. (2017) established a remote laboratory to create a distributed, modular and flexible online learning environment to integrate a set of scaffolding tools and services intended for instructors and learners such as collaboration and visualisation tools, human tutoring and ability for users to share practical sessions. An exploratory study conducted with 139 undergraduate students enrolled in the first year of a CS degree suggested a positive effect of the framework on learners' engagement when they came to practice system administration, and revealed a significant positive correlation between students' activity within the system and students' learning achievement.

In Greece Goumopoulos et al. (2017) addressed distance education challenges through advanced educational material, intelligent tutoring systems, and virtual laboratories. Students engaged in small-scale projects and implemented both software and hardware prototypes.

In Germany a flipped classroom approach was used by Pawelczak (2017) on an elective advanced programming course. Pawelczak (2017) found that students seemed more motivated when they could work with the course material at times of their choosing, and that they were better prepared in the flipped classroom and discussions could be established on a higher level. However, the effort involved in setting up the flipped classroom was very high and course materials had to be updated frequently as programming languages evolved.

Another interesting example from Germany by Troussas et al. (2020) illustrated an intelligent mobile game-based learning application in a HE course to assess and advance leaners' knowledge in the programming language C#. The application employed an assessing knowledge module for testing the knowledge of learners, a recommendation module for proposing personalised collaboration, a dynamic fuzzy logic-based advice generator for tailored assistance to learners' profile and misconceptions, and a cognitive learner modeler that supports the other modules. Troussas et al. (2020) concluded that incorporating personalisation and collaboration in mobile game-based learning can help students increase their knowledge level.

5 Discussion

This systematic literature review used a 3-phase coding process to review 20 selected articles from an initial data search of 231 studies in order to identify common pedagogical approaches, aligned with Education 4.0, that were used in European contexts to support teaching computes science (CS) courses with undergraduate and graduate students. As indicated in results, in terms of RQ1 and RQ2 of the 20 studies conducted in Europe none of the articles explicitly mentioned "Education 4.0". This could have happened as result of the recent conceptualisation of Education 4.0, or due to a lack of adoption of the term Education 4.0 in the specific discipline of CS.

A cluster analysis indicated a three-cluster solution across the 20 European papers, which we labelled as 1) EDU 4.0 light, 2) Project-based/hands-on learning, and 3) Full EDU 4.0. EDU 4.0 light studies mostly had relatively low total Hussin (2018) scores, and often did not include project-based activities. EDU 4.0 light studies mostly focussed on develop independence, anytime / anywhere, personalised, and choice in how to learn. As illustrated by the descriptions of some of these studies, substantial technological and pedagogical innovations were introduced in CS courses, although mostly just focussing on one or two Education 4.0 characteristics. This potentially could be linked to teachers willing to take some innovations forward based upon a particular problem perceived in a course, but mainly "updating" parts of the pedagogy rather than fully redesigning a CS course (Aničić et al., 2017; Author A, 2012a; Mangaroska & Giannakos, 2019).

The second cluster that we labelled as project-based/hands-on learning had a strong focus on project-based learning and hands-on learning. Mostly these studies used collaborative and project-based learning approaches with some interesting innovations, such as where Finnish students were considered as prospective entrepreneurs (Fagerholm et al., 2018). In all six studies there was a strong focus on hands-on and project-based learning, allowing CS graduate to develop strong programming and soft-skills, often working in team. However, due to the nature of project-based learning there was relatively low flexibility in terms of anytime/anywhere, personalisation, and choice of study.

The third and final cluster, Full EDU 4.0, was strongly focussed on hands-on learning, develop independence, personalised, anytime/anywhere, and choice in how to learn. The lowest Education 4.0 characteristic was the student opinion, although this was substantially higher than the other two clusters. Several innovative and integrated perspectives using flipped classrooms (Pawelczak, 2017), game-based learning (Troussas et al., 2020) and online lab work (Broisin et al., 2017) indicated how to help CS students to develop strong project, programming and team skills.

Based on the research articles reviewed in this study we can conclude that Education 4.0 is a new concept in teaching computer science courses and has not yet been utilized by teachers. This study indicated that although this field is at its early beginnings, some basic trends can be noted and conceptualized. In a way it was surprising to identify three distinct clear clusters in terms of design of CS courses. While in some learning design research there is some evidence of common design practices (Author A, 2012a, 2015a; Mangaroska & Giannakos, 2019) when comparing different disciplines, these preliminary findings seem to suggest three broad flavours of design in European CS.

Future research should be carried out to identify and propose corresponding learning designs that would include Education 4.0 characteristics and thus transform the university computer science courses. In addition, it is essential to conduct a wider review beyond Europe to determine whether these three clusters are unique for Europe, or whether similar/different clusters in CS can be defined across the CS field. Finally, it is essential that more research is conducted which skills CS teachers might need to develop, implement and evaluate these Education 4.0 courses, and whether (or) not these courses actually deliver in terms of students expectations and the wider industry.

Acknowledgments

This research/scientific paper was conducted within the project “[blinded for peer review]” funded under Erasmus+ Programme, KA203 - Strategic Partnerships for higher education. The European Commission's support for the production of this paper does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

References

Aghaee, N., & Keller, C. (2016). ICT-supported peer interaction among learners in Bachelor's and Master's thesis courses. *Computers & Education, 94*, 276-297. doi: 10.1016/j.compedu.2015.11.006

Aničić, K. P., Divjak, B., & Arbanas, K. (2017). Preparing ICT Graduates for Real-World Challenges: Results of a Meta-Analysis. *IEEE Transactions on Education, 60*(3), 191-197. doi: 10.1109/TE.2016.2633959

Apiola, M., Lokkila, E., & Laakso, M.-J. (2019). Digital learning approaches in an intermediate-level computer science course. *The International Journal of Information and Learning Technology, 36*(5), 467-484. doi: 10.1108/IJILT-06-2018-0079

Author A. (2012a). [details removed for peer review].

Author A. (2015a). [details removed for peer review].

Author A. (2021a). [details removed for peer review].

Author A. (2021b). [details removed for peer review].

Broisin, J., Venant, R., & Vidal, P. (2017). Lab4CE: a remote laboratory for computer education. *International Journal of Artificial Intelligence in Education, 27*(1), 154-180.

Carrascal, G., del Barrio, A. A., & Botella, G. (2021). First experiences of teaching quantum computing. *The Journal of Supercomputing, 77*(3), 2770-2799. doi: 10.1007/s11227-020-03376-x

Casañ, M. J., Alier, M., & Llorens, A. (2020). Teaching Ethics and Sustainability to Informatics Engineering Students, An Almost 30 Years’ Experience. *Sustainability, 12*(14), 5499.

Charlton, P., & Avramides, K. (2016). Knowledge Construction in Computer Science and Engineering when Learning Through Making. *IEEE Transactions on Learning Technologies, 9*(4), 379-390. doi: 10.1109/TLT.2016.2627567

Cobos, M., & Roger, S. (2019). SART3D: A MATLAB toolbox for spatial audio and signal processing education. *Computer Applications in Engineering Education, 27*(4), 971-985. doi: 10.1002/cae.7

Degener, P., Haak, V., Gold-Veerkamp, C., & Abke, J. (2019, 8-11 April 2019). *Towards the Vision of an LMS Integrated, Browser-Based Simulation to Program LEGO MindStorms EV3s in ANSI-C.* Paper presented at the 2019 IEEE Global Engineering Education Conference (EDUCON).

Dondio, P., & Shaheen, S. (2019). *Is StackOverflow an Effective Complement to Gaining Practical Knowledge Compared to Traditional Computer Science Learning?* Paper presented at the Proceedings of the 2019 11th International Conference on Education Technology and Computers, Amsterdam, Netherlands.

Fagerholm, F., Hellas, A., Luukkainen, M., Kyllönen, K., Yaman, S., & Mäenpää, H. (2018). Designing and implementing an environment for software start-up education: Patterns and anti-patterns. *Journal of Systems and Software, 146*, 1-13. doi: 10.1016/j.jss.2018.08.060

Fisk, P. (2017). Education 4.0 … the future of learning will be dramatically different, in school and throughout life., 20 January 2021, from https://www.thegeniusworks.com/2017/01/future-education-young-everyone-taught-together/

Garousi, V., Giray, G., Tüzün, E., Catal, C., & Felderer, M. (2019). Aligning software engineering education with industrial needs: A meta-analysis. *Journal of Systems and Software, 156*, 65-83. doi: 10.1016/j.jss.2019.06.044

Goumopoulos, C., Nicopolitidis, P., Gavalas, D., & Kameas, A. (2017). A distance learning curriculum on pervasive computing. *International Journal of Continuing Engineering Education and Life Long Learning, 27*(1-2), 122-146. doi: 10.1504/ijceell.2017.080996

Hattie, J., & Yates, G. C. (2013). *Visible learning and the science of how we learn*. New York: Routledge.

Hussin, A. A. (2018). Education 4.0 made simple: Ideas for teaching. *International Journal of Education and Literacy Studies, 6*(3), 92-98.

Jisc. (2019). Education 4.0 – transforming the future of education (through advanced technology). . Retrieved 20 January 2021, from https://www.youtube.com/watch?v=aVWHp8FsV1w

Knobelsdorf, M., Frede, C., Böhne, S., & Kreitz, C. (2017). *Theorem Provers as a Learning Tool in Theory of Computation*. Paper presented at the Proceedings of the 2017 ACM Conference on International Computing Education Research, Tacoma, Washington, USA.

Llorens, A., Berbegal-Mirabent, J., & Llinàs-Audet, X. (2017). Aligning professional skills and active learning methods: an application for information and communications technology engineering. *European Journal of Engineering Education*, *42*(4), 382-395. doi: 10.1080/03043797.2016.1189880

Mäkiö, E., Yablochnikov, E., Colombo, A. W., Mäkiö, J., & Harrison, C. (2020, 10-12 June 2020). *Applying Task-centric Holistic Teaching Approach in Education of Industrial Cyber Physical Systems.* Paper presented at the 2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS).

Mangaroska, K., & Giannakos, M. N. (2019). Learning analytics for learning design: A systematic literature review of analytics-driven design to enhance learning. *IEEE Transactions on Learning Technologies*, *12*(4), 516-534. doi: 10.1109/TLT.2018.2868673

Parejo, J. A., Troya, J., Segura, S., del-Río-Ortega, A., Gámez-Díaz, A., & Márquez-Chamorro, A. E. (2020). Flipping Laboratory Sessions: An Experience in Computer Science. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, *15*(3), 183-191. doi: 10.1109/RITA.2020.3008132

Pawelczak, D. (2017). *Comparison of traditional lecture and flipped classroom for teaching programming.* Paper presented at the Proceedings of the 3rd International Conference on Higher Education Advances.

Salmon, G. (2019). May the Fourth Be with you: Creating Education 4.0. *Journal of Learning for Development*, *6*(2).

Scatalon, L. P., Garcia, R. E., & Barbosa, E. F. (2020, 21-24 Oct. 2020). *Teaching Practices of Software Testing in Programming Education.* Paper presented at the 2020 IEEE Frontiers in Education Conference (FIE).

Schäfer, U. (2019, 8-11 April 2019). *Teaching Modern C++ with Flipped Classroom and Enjoyable IoT Hardware.* Paper presented at the 2019 IEEE Global Engineering Education Conference (EDUCON).

Troussas, C., Krouska, A., & Sgouropoulou, C. (2020). Collaboration and fuzzy-modeled personalization for mobile game-based learning in higher education. *Computers & Education*, *144*, 103698. doi: 10.1016/j.compedu.2019.103698

Urquiza-Fuentes, J. (2020). Increasing Students' Responsibility and Learning Outcomes Using Partial Flipped Classroom in a Language Processors Course. *IEEE Access*, *8*, 211211-211223. doi: 10.1109/ACCESS.2020.3039628

Wasson, B., & Kirschner, P. A. (2020). Learning Design: European Approaches. *TechTrends*, 1-13.