Università degli Studi di Salerno

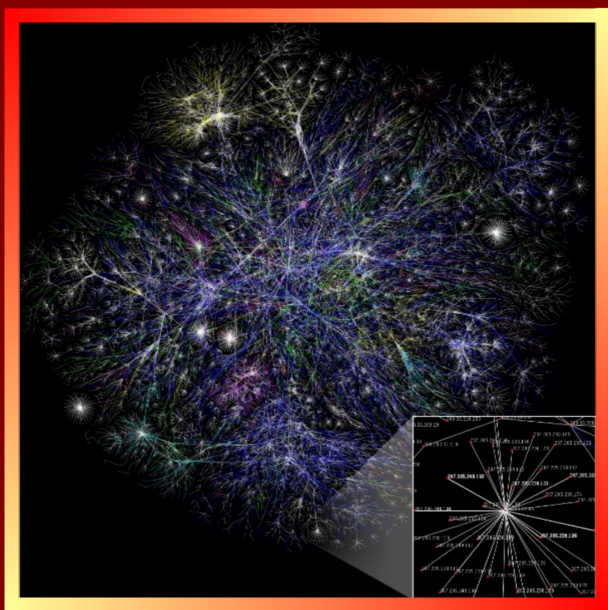
Dipartimento di Informatica

Dottorato di Ricerca in Informatica – XXXV Ciclo

Tesi di Dottorato/Ph.D. Thesis

Outreach in Computing Education: a Design Toolkit

Agnese Addone



Supervisor: **Prof. Vittorio Scarano**

Ph.D. Program Director: **Prof. Andrea De Lucia**

AA 2021/2022

Curriculum Computer Science and Information Technology

University of Salerno

Department of Computer Science

Dottorato di Ricerca in Informatica
Curriculum Computer Science and Information Technology
XXXV Ciclo

TESI DI DOTTORATO / PH.D. THESIS

Outreach in Computing Education: a Design Toolkit

Agnese ADDONE

SUPERVISOR: **Prof. Vittorio SCARANO**

PHD PROGRAM DIRECTOR: **Prof. Andrea DE LUCIA**

A.A 2021/2022

Io sto ancora imparando.
(I am still learning)

— Michelangelo Buonarroti

Dedicated to my son Valerio,
May your learning desire last a lifetime.

ACKNOWLEDGMENTS

A Ph.D. course is a great commitment to research but also a significant part of a learner's pipeline. In my very case, study and life have literally overlapped, intersected and merged with each other. I admit that I have often wavered in persist but, just as often, I received the patient and steady encouragement from my tutor Professor Vittorio Scarano, who pushed me to continue, to see a spark of light beyond every single effort, and not to give up. To me, he represents both an expert guide and committed teacher, but also a brilliant example of researcher to follow and to be inspired by. My heartfelt thanks go to him, I will not forget his advice and I hope he can be proud of my Ph.D. output.

Furthermore, I am sincerely grateful to the esteemed scholars who believed in my preliminary research project and granted me support since the start-up phase: Flavia Marzano, Stefano Converso, Paolo Costa, Roberto Maragliano and -not least- for her academic outreach activity Margaret J. Low MBE, who is an amazing model to many STEM teachers, included me.

Thanks to the ISISlab teachers and researchers, in particular to Maria Angela Pellegrino for her kind and precious support. I warmly greet all my Ph.D. mates at the Lab and Department but also in other universities, mainly Giampiero Dalai for his timely help in *unraveling the tangle*.

Lastly, I am sincerely grateful to my very close friends who never ceased to believe in me. I can not find suitable words for my tireless and greater supporter, Francesca Meloro, who shared my whole Ph.D. journey with authentic friendship.

ABSTRACT

Technology is shaping our lives at an increasing rate and is modeling the way we live in our world. Communications, social media, automation, transactions, video games, are just some of the main purposes in which technologies take form.

In this context, the demand for specific training in the field of Computer Science is growing, and it allows an aware and competent students' attitude in addressing the contents and variety of these technologies.

Today's learners usually face the constantly growing world evolution, its technological progress and the big questions that innovations spark. The increasing demand of computing competences reflects the spread of targeted training to broaden the cultural and technical instruction.

Computing Education represents the discipline and research field that deals with these formative needs, from Computational Thinking to advanced Information Technology literacy, from the development of professional skills to the balancing of the social and economic digital divide issues in order to ensure a civic participation in the society. Computing Education supplies the theoretical foundation to these formative objectives, in order to ensure the achievement of a specific knowledge, skills and digital citizenship.

The school system has begun to respond to these cultural and educational needs from a few decades on, but there is still no widespread teaching on a global level and the consequence is reflected in cultural, economic and social development gaps. Although some countries have started to elaborate school curricula in Computer Science, many others have not yet established formal programs of Computing Education.

In order to compensate for the lack of peculiar pathways, however, many extracurricular initiatives were born and are still widespread, which often act even as a driving force for change in the school. In fact, besides formal institutions and their curricula, the landscape of Computing Education Outreach Programs oc-

curs mainly outside school in non-formal and informal learning environments.

Outreach Programs are those sessions where kids can access computing literacy in a learning environment designed to meet primarily their motivations towards computing and the major education instances from our society.

Even though researchers and educators, as outreach designers, refer to the contents of the school curriculum in the choice of topics and purposes, however these sessions remain complementary to the formal education system in terms of their peculiar features (students' motivation, organization, teaching principles, objectives and strategies) which differentiate them from the school instruction and organization.

The topic of my dissertation fits in the context of the **Computing Education Outreach Programs** addressing primary and secondary school learners, with the aim to describe the design process both in the instructional design stage and in the analysis and resolution of the most common critical issues. In fact, many concerns can suggest designers a set of diverse options when elaborating a program proposal or while re-designing the outreach during the follow-ups.

The contribution of my thesis is to classify **the designers' process features and concerns** by creating a few categories of related issues and areas of influence on learners. Furthermore, I mean to give tangible support to outreach researchers, educators and practitioners in the design or re-design process of the programs by providing them with a practical toolkit to overcome the major issues.

In addition to the specific contribution in the elaboration of the **design process and toolkit**, the thesis deals with some epistemological problems of the subject. In fact, Computing Education is a relatively recent subject that lies between Computer Science and Pedagogy, whose respective experts come from both disciplinary fields by hybridizing and merging their knowledge and skills.

In the dissertation I have taken into account a research method that I can summarize as follows:

1) an *excursus* on the history of Computing Education Outreach Programs and on the major scientific contributions to the inter-

and cross-disciplinary link between Computer Science, Pedagogy and Instructional Design;

2) an analysis and taxonomy I conducted on the research topic, both on the Outreach Programs and the tools, languages, and environments adopted in the outreach practice;

3) a description of the design process and method to detect the major concerns of outreach initiatives;

4) a design toolkit, with guidelines for designers on possible solutions to the above concerns.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ACM Association of Computing Machinery
ADHD Attention Deficit Hyperactivity Disorder
AI Artificial Intelligence
ASD Autism Spectrum Disorder
CEd Computing Education
CS Computer Science
CSEDU International Conference on Computer Supported Education
COVID-19 Coronavirus Disease 2019
CT Computational Thinking
CL Creative Learning
DS Digital Storytelling
FIE Frontiers in Education Conference
HPC High Performance Computing
ICT Information and Communication Technologies
IEEE Institute of Electrical and Electronics Engineers
IT Information Technology
IV International Conference Information Visualization
K-12 Kindergarten through grade 12
MIS4TEL International Conference on Methodologies and Intelligent Systems for Technology-Enhanced Learning
OER Open Educational Resources
OP Outreach Programs
PNSD Piano Nazionale Scuola Digitale
PR Pull Requests
RQ Research Questions
SLR Systematic Literature Review

STEM	Science, Technology, Engineering, Mathematics
TEL	Technology-Enhanced Learning
TLE	Tools, Languages, Environments
TPD	Teachers Professional Development

Part I

INTRODUCTION

1 INTRODUCTION

The progress and evolution of our society, mainly in recent decades, has shown an incremental and pervasive development of digital technologies, which still persists today.

Our connection with some innovations involves us directly in everyday life, in our work, in the education pipeline and in our relationship with others, both as direct users and, at times, as active players.

Many of our interests and daily occupations, which we now take for granted, derive from these creations and require a convenient knowledge of Computer Science fundamentals [133]. We commonly employ some devices for activities such as reading or writing and to perform complex calculations or to represent data. Access to the Internet is universally considered an indispensable means of collecting information but also guaranteeing our activities, communication and social interaction.

In our homes and wearable tools the voice assistants are already a consolidated reality and they represent an indispensable support to the inclusion of impaired people. The entertainment industry benefits from Information Technology (IT), for example in video games development or cinema production. The access and connection with the public administrations are commonly mediated by websites and applications, provided by systems of digital identity.

Therefore, technology and computing acquire a crucial value as agents of change, inclusion, development and progress in each of these areas and in many others. They can facilitate the growth of people's knowledge, and their attitude towards the continuing education, even across several disciplines or fields. Their acquisition can also lead to the elaboration of useful skills for study or work [140].

Some technologies also represent the means to the empowerment of individuals and the access point for their civic participation, while reducing the digital divides and increasing citizens' self-determination [62, 133]. The major and more evident impact of this process can be observed especially for school students, women, and those people who are often underrepresented because they live in disadvantaged socio-economic backgrounds: the scope of a popular digital literacy is to support these people to become active and engaged participants in public life [166], but also responsible users of technology with suitable knowledge and skills.

In some cases, in fact, our perception of the pervasiveness of several technologies can be distorted by a lack of awareness or knowledge, and we can become passive subjects of some applications of them or even of deliberate actions performed by third parties by using the influence of platforms or tools.

Sometimes we can hardly identify the actions carried out towards, for example, our privacy data or even the risks for our own identity, and how they can compromise our digital security. In some cases the impact of an improper and malicious technology utilization can even affect our opinions or behaviors with fake news on crucial matters like public health or politics, for example. They can have a snowball effect on our capacity to paying attention to online risks and on our ability to develop awareness in relation to civic matters, on the right to access and participate as citizens in public life, on the spread of hazardous or false contents.

These aspects become even more crucial when they involve the younger generations, who can ignore the positive or negative effects of new technologies, because they have not yet had the educational, cultural and social opportunities to develop a full awareness of the risks and opportunities.

In the educational field but also in our mass culture, a rather widespread opinion circulates that the younger generations are naturally already computer literate, as if their experience were a consequence to birth and not rather the outcome of a process of construction of learning.

This supposition, which also gave rise to definitions such as that of *digital natives* [121], has for a long time generated an educational misunderstanding about considering barely important a specific education in computing. Currently instead, it is intended functional and transversal to all disciplines, more as a means than as a purpose. Reality then revealed the inconsistency of the above definition and research has shown that this scientific and technological education has its proper foundation, so that it should be opportunely integrated into the school curriculum and pipeline since childhood.

The sudden and continuous technological development faces adults and kids through some educational matters, such as gradually constructing an efficient and aware digital literacy in order to untangle themselves from the obstacles and to critically evaluate both the risks and the opportunities.

In relation to age but also to cultural and social background, today's students need to learn the basics to actively interact with this fast digital growth. At the same time, a critical attitude towards technology becomes their passport to consciously participate in the present time instances while addressing the future concerns. School, culture, the construction of a work activity, civic and political participation of all components of society with their respective differences and peculiarities, economic progress and environmental sustainability are just some of the themes that characterize and can also be addressed with a critical eye and conscious use of technology [19, 51, 66, 133, 156, 166, 167].

Therefore, researchers and teachers are the ones who play an important role in supporting students to develop this attitude and to face the ongoing technological progress by adopting wise and thoughtful solutions. At all school grades and for every audience or target, many questions and matters raised by innovations enhance the demand of computing knowledge and competences.

In response to this powerful and constant cultural prompt, from decades several national school systems and public institutions have been developing digital education and computing literacy policies [14, 15, 33, 35, 41, 69, 112]. The educational choice to spread scientific and technological knowledge to all learners, which mainly concerns the public school, is accompanied by many extracurricular campaigns led by universities, foundations, private subjects or volunteering organizations for the promotion of Science, Technology, Engineering, Mathematics (STEM) disciplines among young generations, from childhood to adolescence and up to the university choice [10, 61, 71, 98, 108, 117, 122, 127, 141].

In parallel, scientific research has explored these initiatives, providing studies and surveys that analyze methods, topics, tools and the educational impact of information technology in education, giving birth to a peculiar field of investigation.

Computing Education (CEd) is this discipline and research field. It addresses the theoretical basics of the above cultural, social and economic issues and their formative objectives; it analyzes how to ensure to every learner a specific knowledge background in computer science in relation to their aspirations and needs; it evaluates the construction of digital competences for education, life and citizenship. Furthermore, it is the main topic in which this dissertation fits.

While institutions are developing computing frameworks and curricula, adapting them to every school level and enhancing the teachers' professional development, many other programs are led in extracurricular environments. My thesis will address and concern the research on these initiatives, which go under the name of Outreach Programs (OP), led outside or beside the formal school curriculum in different learning environments.

1.1 RESEARCH PURPOSE

The research area of my thesis, namely Computing Education, is located at the intersection of two main disciplines, Computer Science (CS) on one side and Education on the other. Each of them contributes with its own characteristics, methodologies and tools to generate a further field of investigation that is increasingly becoming transdisciplinary.

In order to build Computing Education, research in CS and STEM disciplines joins together with learning sciences, instructional design and social sciences. Until a few decades ago scholars, who belonged to two distinct disciplinary areas, approached the subject according to their respective fields and points of view. It has been a long journey to match their diverse principles and to reach a common synthesis taking into account every aspect of the new discipline. Today these scholars are usually involved in a dialogue and they support each other to compare and validate their respective approaches [148].

Computing Education faces the educational challenges of our era with the aim to grasp the demands from society, culture, school and technology [45]. Its field of investigation is diversified and takes advantage of the contribution from scientists with very different backgrounds who collaborate to hybridize their respective knowledge and skills.

This thesis deals with one of the investigation areas of Computing Education, i.e. Outreach Programs.

As seen above, institutions and schools have promoted digital and information technology education and still outline frameworks and guidelines to incorporate information technology into school curricula.

Since these initiatives naturally have a gradual and rather long development of integration with other disciplines, but above all since they are not widespread on a global level, in parallel many extracurricular courses promoted by other entities, precisely the outreach, have also taken shape.

Outreach Programs are the sessions, short courses and camps held outside the school curriculum and the formal education organization, designed to foster digital awareness among young people from an early age. They are often promoted by institutions, as schools and Universities, but also by private companies and associations, to address computing to complement the school curriculum.

The contribution regards, and is limited to, the initiatives addressing K-12, i. e. kids and students ranging from primary up to the end of high school. Concerning students enrolled in a school path, I have also studied which aspects of Computing Education influence their career choice and access to a post-secondary pipeline in STEM, and the role of teachers and educators in the process.

Outreach Programs in Computing Education take place in non-formal or even informal learning environments, meaning that they coexist besides school and formal agencies, and they also represent an original *approach to learning*. In these sessions young people are actually active learners, and they are supported in the computing literacy process by teachers, educators or simple volunteers who facilitate their participation.

The environment is designed on the basis of an intense preliminary research work on their motivations and expectations towards information technology, but also with the scope of bridging the digital knowledge that characterizes our society.

Outreach deal with these formative requests, ranging from Computing Education theoretical foundations, such as those of computational thinking, to sessions aiming at advanced computing literacy; they aspire to develop the skills required in the most varied professional fields; they keep an eye on the dynamics of civic participation and study how to bridge and balance digital, gender and social divides. Furthermore, in designing outreach, the researchers aim to guarantee equity in access the initiatives and the inclusion of all the subjects involved [58], independently from their personal, social or economic backgrounds.

One of the main Outreach Programs purposes is to improve and increase students' participation in Computing Education, develop digital skills, promote online participation and awareness, expand creativity in learning. Outreach Programs represent a broad field, with a huge variety of courses and camps for learners, whether students or teachers. Therefore, they also aim in parallel to improve the methodologies and the choice of Tools, Languages, Environments (TLE) to ensure the advancement of the Teachers Professional Development (TPD), so K-12 educators can support students both in and outside the classroom.

Other Outreach Programs challenges are meant to actively engage kids and encounter their aspirations, in order to promote the transition from school to STEM careers. Many false beliefs and misconceptions can affect the opinion that learners can have previously regarding technology and related fields, and they are due to social and cultural bias that I will explain in the thesis.

Outreach have also represented, in very recent times, a solution to face emergency situations, by designing new educational environments and communities, as seen during the Coronavirus Disease 2019 (COVID-19) pandemic. Many activities had to suddenly switch to an online modality and teachers benefited from outreach experiences to develop new teaching methods.

1.2 DISSERTATION CONTRIBUTION AND METHOD.

My thesis topic is therefore the Computing Education OP for primary and secondary school students (also called with American connotation K-12), with the main objective of describing the **design process** and implementing a **toolkit** to support researchers, educators, professionals and practitioners to overcome the main problems. In fact, many concerns can suggest designers a set of diverse options when elaborating a program proposal or while re-designing the outreach during the follow-ups. The research idea is to provide them a tool with several guidelines of intervention.

Hence, my dissertation grounds on two main Research Questions (RQ)s:

- **RQ1** - What are the major design concerns of an Outreach Program in K-12 Computing Education?
- **RQ2** - What are the possible solutions to pursue for ensuring efficacy to the Outreach Programs?

In reply to RQ1, I intend to describe the designers' process and concerns by creating a few categories of related issues and areas of influence on students' learning or behaviors; by answering RQ2 instead I mean to provide tangible support to researchers in the design or re-design of the programs by equipping them with a practical toolkit.

In addition to the specific contribution in the elaboration of the design process and toolkit, the thesis deals at a theoretical level with some epistemological problems of the subject [148]. As seen above, Computing Education is a relatively recent subject that lies between Computer Science and Pedagogy, whose respective experts strive to cooperate and merge their methodology and knowledge [148]. The coexistence, discussion and collaboration between different disciplines is of vital importance for understanding the possibilities and boundaries of the study. Only from the collaboration between scholars of different disciplinary fields in recent decades it has been offered a profitable and progressive research progress.

My research method can be summarized in four main steps:

- a Computing Education *excursus*, from its history to the major scientific contributions, resuming some of the interand cross-disciplinary links between Computer Science, Pedagogy and Instructional Design;

- a survey on the OP in K-12 CEd [5] and a taxonomy on the TLE [6] that I conducted regarding the research topic;
- a description of the Outreach design process and the method to detect the major concerns;
- a design toolkit, with guidelines for designers on possible solutions to the above concerns.

Even though researchers and educators, as outreach designers, refer to the contents of the school curriculum in the choice of topics and contents, however these sessions remain complementary to the formal education system in terms of their peculiar features (motivation, organization, teaching principles, objectives and strategies) which differentiate them from the school instruction and organization. So they need to adopt different solutions regarding every planned activity.

Then I will provide a proposal of a design toolkit to support OP designers in K-12 Computing Education, both in the instructional design stage and in the analysis and resolution of the most common critical issues.

I will examine the design of the initiatives already carried out, as they result from a review of the literature I conducted on a defined query on Outreach Programs in K-12 Computing Education [5]. The survey made it possible to categorize the OP along their peculiar aspects, but also to discover and determine some design issues that challenge researchers. I will also focus on their impact, both singularly on different audiences and on the wider influence that they can generate on the transition from school to University and the choice of STEM careers.

The scientific contributions that I had the opportunity to present and publish during my Ph.D. path range from OP to TLE, to Digital Storytelling (DS), to Computational Thinking (CT) and Creative Learning (CL), to Technology-Enhanced Learning (TEL) (Table 1.1). The papers and articles about OP, TLE and DS found the Chapter 3 where outreach are mainly illustrated.

Table 1.1: Peer-reviewed publications supporting my dissertation

TOPIC	YEAR	PUB. TYPE	TITLE	REF.
OP	2022	Conference CSEDU	Outreach in K-12 Programming: A Systematic Literature Review on Audience and Purpose	[5]
OP	2023	Conference FIE	Towards design guidelines for K-12 Outreach Programs	under evaluation
TLE	2022	Conference FIE	Towards a collaborative taxonomy of Tools, Languages and Environments in K-12 Computing Education	[6]
DS	2020	Conference IV	Visual Storytelling by Novelette	[2]
DS	2021	Journal IEEE Access	Novelette, a Usable Visual Storytelling Digital Learning Environment	[3]
DS	2021	Conference MIS4TEL - workshop TEL4FC	Engaging Children in Digital Storytelling	[4]
CT	2021	Conference Proceedings Book chapter	Chesscards: Making a Paper Chess Game with Primary School Students, a Cooperative Approach	[1]
CL	2014	Conference DidaMatica	L'apprendimento creativo dell'informatica in contesti informali: l'esperienza di CoderDojo	[122]
TEL	2022	Proceedings MIS4TEL	Methodologies and Intelligent Systems for Technology Enhanced Learning	[123]

1.3 MY MOTIVATION TO UNDERTAKE A PH.D.

Before undertaking the Ph.D. path in Computer Science at the University of Salerno, in my job as primary school teacher, Computing Education was a topic and practice that I worked on for many years.

At the beginning of my work, my personal interest for Computing led me several times to experiment with my students some activities and solutions to better involve them in STEM. As my experience matured, I realized that in reality we were both just stuck on an instrumental use of some computer applications.

This reflection corresponded not only to my demand for professional improvement, but also to the requests from the children to go further in understanding and gaining knowledge, while becoming active subjects of what they were learning and creating.

Over time, however, supported by a journey of professional development training and a supplement of autonomous self- and continuing learning, I had also the opportunity to explore students' motivation in becoming active creators and novice programmers, and what gradually led them to undertake a pipeline of further studies and specialization in the STEM disciplines.

Computing Education is still not mandatory or included in the curriculum of elementary, middle and high school in many countries. Before its introduction in the Italian school with the *Piano Nazionale Scuola Digitale* (*PNSD*), i.e. the Italian National Digital School Plan, I decided to take a step forward by studying what was already happening informally outside the institutional context. Finally, I decided to embrace a participation in *CoderDojo*, which is an informal movement of coding clubs for kids, led outside school by volunteers [122]. The design and organization of these events was intense because I had a role as champion both for my town chapter and for the whole Italian network, for which I acted as spokesperson and disseminator.

The discussions and dialogue with children and young people, but also with educators, parents, computer scientists, representatives of culture and institutions, has led me to frequent reflections on the impact of these initiatives, but also on the role of teachers in children's learning and creative process.

Designing a program for everyone, paying attention to different ages and aspirations; acting inclusively so that no one would be left out; coordinating the available human and technological resources; it represented not only a training ground for me as a person, but above all for the teacher I was striving to be.

When I started spreading publicly our activities, I realized that, in order to be understood by people, it was necessary to make coincide the outputs of many initiatives with the world of schools. We operated in an informal environment outside the school, which effectively allowed primary and secondary school students to approach information technology in a fun, creative, non-judgmental way.

For this reason, I decided to put myself at stake and resumed studying to systematize what I had learned from my experience in coding clubs. So I realized that my insights and intuitions would find a more suitable place in a university research path, and I decided to undertake a Ph.D. My initial research project and subsequently the work on my thesis therefore represent the result of the journey that has brought me here and which combines two aspects of my training, in school and in volunteering.

1.4 DISSERTATION CONTENTS

The thesis work kicks off in Chapter 2 from an introductory picture, of pedagogical and cultural nature, on the history and theoretical principles that guide Computing Education and related research in the context of the training of students and teachers in primary and secondary school. The most important studies are analysed, and in particular those which constitute an important turning point for the research on the subject.

In Chapter 3 instead I trace the state-of-the-art of the specific research on Outreach Programs addressed to the K-12 grades. The insight grounds on a Systematic Literature Review (SLR) I conducted during my Ph.D. course. Furthermore, a taxonomy of the most common outreach Environments, Languages and Tools is presented, which also led by consequence to the prototyping of a collaborative platform, open to researchers from all over the world to welcome contributions and comments. Last, I will report an outreach and co-design experience planned and performed in our research lab and in schools.

In Chapter 4 I deepen and reflect on the design process by topics and concerns emerged from the analysis of the SLR on the international initiatives already tested and implemented in practice, presented through scientific contributions and already published and validated.

Finally, the analysis of the matters connected to the outreach design led me to the elaboration in Chapter 5 of a toolkit of some effective solutions to the problems most encountered, due both to the type of initiatives and, in some cases, the lack of a specific curriculum in which insert them. The review is a proposal of educational framework to guide researchers and educators in the planning based on the previous practice, the cultural and social instances, the availability of infrastructures.

Part II

THE BIG PICTURE

Perhaps if we wrote programs from childhood on, as adults we'd be able to read them.

Alan Jay Perlis, *Epigrams in Programming*

2 THEORETICAL OVERVIEW

The relatively recent birth of Computing Education and its dynamic evolution correspond with the progressive state of the related research, both in IT and in Education, but also with the advancements in Computer Science and the innovations in technology.

A further definition of K-12 Computing Education has had a development over the last twenty years, with studies ranging from school to university training, to TPD.

In this chapter I will present the main lines of the discipline, especially referring to the research on teaching and learning for the K-12 audience, which represents one of the possible fields of application of methods and practices.

2.1 TERMINOLOGY

There are some specific regional differences in the terminology used in Computing Education. By surveying the scientific literature we deduce that some terms vary from country to country and, since in this thesis I have chosen to adopt some words to define specific targets, audiences, methodological guidelines and teaching practices, I considered appropriate to draw up a brief terminology before addressing the topic.

Inter- and Trans/Cross- disciplinary - The natural contribution of different disciplines to the same research topic might vary based on how they contribute to the topic itself. When the various fields are involved and maintain their respective peculiarities, we speak of inter-disciplinarity; if, on the other hand, the contribution of the different research areas must lead to a synthesis and the disciplines with their contents and methodologies merge, then we speak of trans- or cross-disciplinarity.

Computing Education - Under this definition we find a cross-disciplinary research and didactic field of Computer Science and Pedagogy, which studies and design educational experiences, methodologies and frameworks for teaching and learning Computer Science in the classroom or outside.

K-12 Education - The notation K-12 is genuinely from the US school system and refers to the grades from K (Kindergarten) to the 12th, meaning the audience from primary/elementary grade up to the end of high school and before the post-secondary grades. In recent years many CEd researchers, also non-US natives, adopted it *per antonomasia* or as a synonym of that audience. The definition is employed in studies in which there is an homogeneous approach to methods, due to the age range and the school grades of the subjects involved.

Formal, Non-formal, Informal Education - Learning does not take place exclusively in the school context, but sometimes it can take place outside of it. It can have characteristics related only in part to school or that are even different from it. In its various forms, it therefore distinguishes between a formal, non-formal or informal approach. Formal learning is closely linked to the school, its organization, context and curriculum. We define non-formal the activities related to the school curriculum but which take place in venues and forms that are complementary to the institutional ones, often as enrichment (outreach) initiatives. In the last case, informal learning defines other methods of approaching learning, often completely unstructured and unconventional, closely linked to the spontaneous initiative of the learners, to platforms and tools which are not directly linked to a defined educational framework but which can be used or even constructed and driven by the learner.

Outreach - The definition of outreach includes all the activities and initiatives for the enrichment, expansion and improvement of others that take place in well-defined and often structured contexts, or even the less formal environments that adopt innovative organization and methodologies to convey educational content.

STEM - Over the last few decades this acronym, born to resume four disciplinary domains (Science, Technology, Engineering, Mathematics) has come to indicate the training environments with homogeneous characteristics in terms of approach, methods and research. There is a common thread between these disciplines but also in the way students and teachers approach to them.

Instructional Design - This term indicates the complex of research, design and practices related to the transfer of knowledge. In fact, it deals with the planning of educational experiences, their implementation, the subjects involved and the teaching approaches.

2.2 AN INTER- AND TRANS-DISCIPLINARY FIELD

In the early days, researchers in Computing Education had a background in Computer Science and mainly post-secondary CS; later, scholars of Education joined them, and the osmosis between two areas of research began, determining a cross-disciplinary contamination [45, 56]. Their respective vocabularies, epistemologies, methods and achievements [8] have carried out a reciprocal action of influence until reaching new syntheses.

Nowadays, related fields interact by sharing their principles and strategies in developing curricula or designing courses; the work of the researchers is permeable and open to the comparison and expansion of the characteristic boundaries of the single disciplines.

It is also important to add that Computing Education has increasingly expanded and that we currently find university departments in which this discipline is expressly addressed under this notation. This tendency highlights both the value and the commitment attributed to the representatives involved in implementing these courses and also how important is the relationship with other educational agencies.

The same CEd spread is happening in schools, were the influence of computing on other disciplines is becoming more and more crucial; teachers feel they are more frequently involved in meeting and solving the educational needs of students living in a constant transformation and technological acceleration of the society [140].

This dissertation fits in this part of the theory of CEd, with the aim of adding a contribution to the segment of research related with both the social sciences, pedagogy in particular, and instructional design.

Indeed, related works on K-12 Computing Education programs highlight the role of design in planning initiatives that promote crucial skills such as problem solving or computational thinking [113, 118].

In the context of Computing Education, some researchers highlighted the main lines of research currently underway [16, 17]. They can be summarised as follows:

- Labor market skills;
- Computational Thinking;
- Computational literacy;
- Equity of participation.

Labor market skills. Skilled workers in IT are increasingly required by the labor market. The more and more specialized professions deal with the major technological issues of our society. Industry and the progress of innovations and technologies, like the Artificial Intelligence (AI) systems for example, are creating new professions and roles which need CS advanced literacy.

Computational Thinking. The researchers study the computational ways of thinking and applying computing to other disciplines. Computational Thinking is considered the educational foundation to problem solving and the first step to learn to program in school grades.

Computational literacy. When the basic literacy in CS is done, more complex or specialized concepts appear, and learners need a more advanced construction of their computing knowledge and skills.

Equity of participation. IT knowledge lead to the acquisition of civic participation and understanding the functioning of our society and the main technologies. The involvement of citizens in CS represents an enabling factor that guarantees a fairer society with equal opportunities, regardless of the context of belonging of the individuals.

The first reflections on the importance and need to create an interdisciplinary link between Information Technology and Education date back to the 1960s, when Alan Perlis stated[1] that

1 Lecture at the "Computers and the World of the Future" Symposium, Massachusetts Institute of Technology (MIT), 1961.

programming should be part of the liberal education and suggested to create specific computing courses at the university level [55].

Later, other brilliant researchers' insights [44, 115], as we shall see, have contributed to further develop the theory of Computing Education starting from childhood on, and pointed out their attention on the importance of programming in Education, the development of educational languages and the role of learning environments.

The limit of these studies and related experiments was in the simultaneous development stage of hardware and software, which still did not allow for a profitable user experience. Programming languages were essentially text-based and novices struggled with command translation and conceptualization [57]. The first experiments concerned research on the difficulties that novices could have in programming and how to design effective experiences in a targeted way for those approaching computing, and mainly programming, for the first time. Therefore, researchers also began to study the behavior psychology of those who had no rudiments of the discipline and encountered the difficulties of processing computational thinking.

The first computer language expressly designed for educational purpose, i.e. LOGO, dates back to the 1967 and its designers intended it as a conceptual framework for teaching mathematics and mainly problem solving to novices [44, 144, 152] (Figure 2.1).

During the same period we find the first experiences in teaching and learning computing to -and with- children, like the experiments that Seymour Papert conducted with novice kids [115] or later Alan Kay to engage them in arts and music [74].

Since then, to mitigate the difficulties of programming, novices were also assisted by, and could interact with, some physical tools [57], from the pioneering tangible LOGO buttons instead of the text-based turtles [120], to the LEGO building sets [126], to block-based programming languages like Scratch [127].

In many studies, from Papert's *Mindstorms* on, Computing Education opts for a student-centered learning design and emphasizes the active role of the learner in the process.

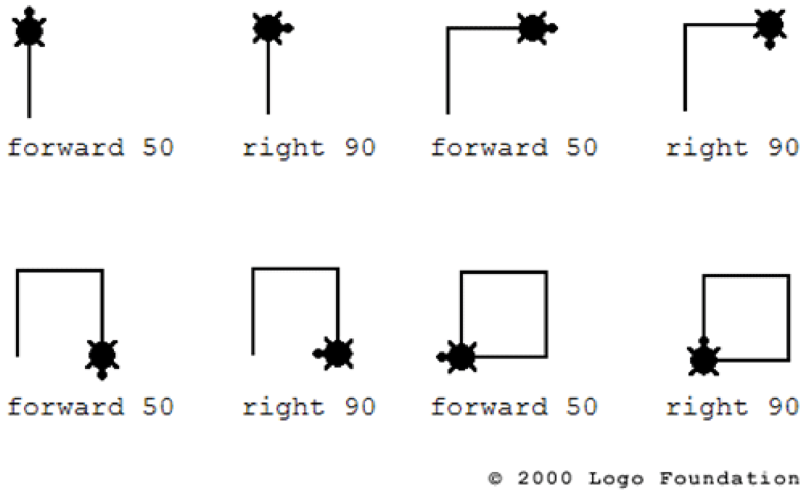


Figure 2.1: The LOGO turtles.

The efficacy of Computing Education is also related to the role of positive incentives or gratifications, and to the deconditioning from family and social context.

In one of his well-known reflections, Seymour Papert noticed that "in order to ensure a real acquisition of knowledge and to enhance motivation by providing a positive feedback, the design of the environment and context should be meaningful for the learners and close to their significant life experiences and circumstances" [115].

2.3 BACKGROUND STUDIES

The research on Computing Education is very extensive but it is nevertheless possible to highlight some general works that deal with this area more broadly and provide the foundation for further study.

In particular a large comprehensive volume [45] delineates an overview of the CEd research and its branches, addressing the main themes by which the researchers are challenged. The book contains the contributions of many scholars who represent an international reference on these issues.

Another very interesting and useful work to deepen the state of research in K-12 Computing Education, even if mainly focused on teaching programming to K-12 children and teens, is a Handbook [56] with many interesting contributions from scholars and experts in the field.

The third inspiring contribution is a volume published very recently [140], which in its first part outlines some important innovations in educational research and in the epistemological foundations of CEd.

From the analysis of the literature, two main research tendencies emerge: the first one is more oriented towards considering CEd as a category of mathematical logic and computational thinking; the other one instead is more based on the design of educational experiences, the interaction between educators and learners, the use of tools and languages, and the construction of artifacts [152].

In relation to the four areas of CEd research mentioned above, there are many contributions of a more specific nature on single subjects of the discipline. I have focused in particular on initiatives to widen participation [81], to disseminate programming [44, 92] and finally on Outreach Programs, which I will broadly discuss in Chapter 3.

2.4 COMPUTING AND SOCIETY

The pervasive nature of technologies and their impact on our lives are perceived in every sector, but mainly in Education and in the subsequent transition to the job field.

Nowadays it is complex and challenging to effectively determine the role and impact that technology has in our lives as users, and some important consequences deriving from technologies risk to being overlooked [34, 66, 133]. A guided and critical approach to dealing with Information Technology with the methods of instruction and training therefore becomes decisive.

It is evident the growing commitment of research but also of governments in making IT courses increasingly accessible to all students, to guarantee them the achievement of knowledge, skills and opportunities [33].

For this reason, curricula for formal CS education are being developed in few countries [14, 41, 69, 112] with also many extracurricular initiatives for achieving knowledge of the fundamental aspects of the discipline, its potential applications in real life contexts, the opportunities it can offer for job access and civic participation, and the limits or risks it can prompt [10, 61, 71, 98, 108, 117, 122, 127, 141].

The main interest of research in Computing Education, which operates with an inter- and cross-disciplinary method combining Information Technology and Pedagogy with Instructional Design, is to support and guide governments' decisions on a rigorous and scientific basis.

Computing is also an access key to many other fields of knowledge and those who are not trained in this discipline must have the possibility of approaching it at least with an aware, guided use.

Many peripheral places are affected by the lack of infrastructures, as well as devices and adequate training opportunities; some segments of the population live in a condition of social and economic disadvantage and are unable to access technology and computing education because they lack the operational and cultural tools. Finally some groups, for reasons related to gender, ethnicity or even disabilities remain excluded from this type of training course, and the initiatives specifically designed for their full integration are necessary for them.

2.5 COMPUTING EDUCATION AND COLLABORATIVE LEARNING

An extremely significant aspect in Computing Education is the drive towards collaborative learning, teamwork, the formation of more or less homogeneous work groups based on the belonging to the same audience or - for example - to a segment of the society.

In addition to being effective, it is also inclusive because it enhances the skills of each participant, highlighting their strengths and resolving their weaknesses. Another important contribution to research, deriving from the theme of collaborative learning, concerns the role of the community as a factor of inclusion and a driving force for learning.

2.6 COMPUTING EDUCATION AS A FACTOR OF EMANCIPATION

Computing education, especially in non-formal and informal environments is often used to involve the social groups who live in cultural, social and economic disadvantage and sometimes, for various reasons, even outside the school context.

The fundamental active involvement of some groups, such as children of ethnic clusters who live at the margins of large industrialized societies, or those who reside in geographically inaccessible places, or those who have suffered serious economic or cultural deprivation because they come from families with poor belief in the school system or legality for example, becomes the starting point for their social redemption [66].

Young people are invited to get involved and opportunities are created to broaden their cultural horizons but also to create a course of study, or deepen a passion that could lead them in the future to create their own profession, freeing themselves from an inherited and not chosen status.

Computing Education, in its purpose of enabling and emancipating factor, dialogues with the social sciences, mainly when talking about children and teens [19, 133, 156]. As we shall see, designers consider the students' life context (family, school, peers) and their cultural, social and economic background to be very important.

In support of these reflections and plans, we find a few studies on critical pedagogy, which is the cultural and philosophical movement that mainly address the effective abilities of learners, how to promote their active responsibility towards technology and the opportunities or risks of the web, in view to determine their future as active and responsible citizens [51, 166, 167]. Some of these studies approach technology from a sociological point of view and often deal with topics apparently not directly related to school practice; however they deepen general topics on the role of some technologies in our lives, on how they are designed and how they should be regulated [34].

The implications of a social nature reflect the dynamics of our society and scholars highlight strengths and weaknesses on which it is possible to intervene: from the design of inclusive AI systems, to the protection of privacy, to the limitation of the surveillance by voice assistants, to social media algorithm bias, and much more.

The implications of this critical thought are also of a political nature [51], but they undoubtedly give rise to decisive reflections on the role of technology in our society and how education should be built.

In some context, K-12 Computing Education is facing student engagement but also attempting to prepare educators with critical pedagogical frameworks in order to design social justice [66].

Part III

THE OUTREACH PRACTICE

No one is born fully-formed: it is through self-experience in the world that we become what we are.

Paulo Freire, *Pedagogy of the Oppressed*

3 OUTREACH IN K-12 COMPUTING EDUCATION

Outreach Programs in Computing Education represent a significant variety of courses and camps for learners, whether students or teachers. They are unconventional initiatives often promoted by schools or university CS departments, but there are also many programs planned and promoted by private companies or civic associations led by volunteers.

Although in literature there is not a global unique definition of what K-12 Outreach Programs are, we can include among them the non-formal or informal initiatives that implement or enrich the students' school curriculum or even the TPD.

OP promote a different point of view, a given methodology, or a new technology that can have an impact on learners' future choices towards CS or, more broadly, STEM disciplines.

The pedagogical approach adopted boosts learners' active or sometimes self-directed learning, creativity, collaboration and peer mentoring while removing social, cultural, emotional barriers and stereotypes about the discipline.

Outreach Programs attenuate formal roles or hierarchies between teachers and students, and create a learning environment less teacher-centered than in schools, where kids can hold and somehow drive their own knowledge with a more intentional attitude. Hence, OP are not directly based on the institutional curriculum or a computing framework and do not usually entail the traditional forms of evaluation, opting for self-assessment and community discussion on error resolution.

Outreach Programs are definitely complementary to formal education, they promote a constant experimentation of tools, environments, scopes and methodologies [12], and tune in with the instances of social construction of knowledge of our era.

In support of the topic of my dissertation and to develop the main passages of my research on OP, I will present in this chapter mainly some specific publications, i.e.:

- a SLR on OP;

- a taxonomy on TLE;
- an outreach experience on storytelling and OP design in K-12 CEd.

As a starting point of my course, I conducted a SLR on Outreach Programs [5], narrowing the query to those in Computing Education for the K-12 students. The survey outlines the features and principles of the programs, with some remarks on the main results.

From the survey I derived in parallel another review of the main TLE adopted in OP. I gradually surveyed the most popular in CEd and, during the collection, I created a taxonomy, from which later a public collaborative platform was born, open to the contribution of other researchers [6]. We are now working on an extended version of the paper, which has been selected during the FIE 2022 Conference as a high quality research paper for the next IEEE Transactions on Education (ToE) Special Issue "Engineering Education Beyond the Pandemic" in 2023.

In this chapter I also report a significant outreach experience in which I participated, during the development of a visual digital storytelling platform which we carried out in our research laboratory [2–4]. It represents one of the main factors that led me to reflect on some aspects of outreach features and design, in particular on the engagement of participants and the co-design of the learning environment.

3.1 MY RESEARCH BACKGROUND AND EXPERIENCE

Before addressing the topic of my dissertation, I would like to outline the spark which drove me here.

The research project that led me to embrace a Ph.D. course and later to write down this dissertation grounds on a background in Computing Education which is both the sum of my job and of several outreach activities I conducted along the years. I will resume here what I already anticipated in Chapter 1.

School - I am a tenured teacher in the Italian primary school, so I definitely work in the K-12 segment. Technology is part of my everyday activity with kids since my first steps in Education.

Computing has characterized our lessons along the years until a decade ago, when I started introducing the educational use of Twitter for literature re-writing purpose[1].

At that time, me - the kids and me - were just users of software and devices, adapting them to other activities. So I felt that in my job something important was missing and I decided to start a personal path of professional development in CEd. Unfortunately, my role as learner was not satisfactory, and for this reason I studied how to move a step forward to improve and upgrade teaching and, by consequence, learning Computing.

CoderDojo and programming clubs - Thus, in 2013 I embraced an international network, CoderDojo, in which volunteers facilitate programming to kids aged 7 up to 17 in informal coding clubs. Once again I was with K-12, but in an outreach environment. For three years my participation was very active both in organizing sessions and in divulgation, as I was leading my town chapter in Rome, Italy, and co-leading the Italian network.

In many public events I was spokesperson of the movement and I disseminated the CoderDojo's vision on how to support kids in becoming active creators of technology instead than passive users[122].

Among others, in 2014 I was speaker at the first Barcamp on the Italian Digital Agenda[2] and at the only session with kids at the Italian Parliament[3]; in 2015 I also spoke at the European Commission in Rome and at the Internet Governance Forum hosted by the Italian Parliament.

In 2014 I was one of kids' chaperons at official events at the European Parliament during the European CodeWeek[4] and at the Coolest Projects [124], which is the main happening for sharing kids' creations with peers.

1 A report of the project on the use of tweets and Gianni Rodari rhymes with primary school kids is hosted by the LTA - Audiovisual Technology Lab of the University RomaTre led by Roberto Maragliano: https://ltaonline.wordpress.com/2014/03/17/twifavola-una-scuola-primaria-gioca-con-rodari-in-140-caratteri/ (Italian version only, last accessed 2023).

2 https://www.camera.it/leg17/537?shadow_mostra=23934; last accessed 2023.

3 https://www.camera.it/leg17/537?shadow_mostra=23991; last accessed 2023.

4 https://blog.codeweek.eu/coderdojo-is-coming-to-the-european-parliament/; last accessed 2023.

Scratch - The Italian coding clubs early adopted mainly Scratch as a programming language and safe platform for K-12 kids. The Scratch environment was designed at the LLK -Lifelong Kindergarten group at the MIT MediaLab, led by Mitchel Resnick, intentionally for an educational purpose [127]. The Scratch community was another amazing encounter for me and an opportunity to upgrade my pedagogical knowledge in CEd: I met the Director and team of LLK, but also several educators and scholars from all over the world. With some of these amazing scholars, educators and professionals I constituted a Program Committee and finally organized two European editions of the Scratch Conference in Amsterdam (2015)[5] and in Bordeaux (2017)[6], where we gathered the international community of Creative Learning. With Scratch and its community I had the opportunity to study and deepen the technological vision and pedagogical figure of Seymour Papert, learning a lot on the Constructionism theory[115, 116, 146].

Consulting - The above outreach activities made me develop communication and managing skills for organizing educational events, but they also gave me the knowledge to design CEd outreach experiences for, and with, kids.

Indeed, later I started working as K-12STEM consultant for big events such as the MakerFaire Rome [93] or the Rome Video Game Lab [131], the annual festival on gaming taking place in Rome.

Currently, I am involved in the Laboratorio CINI Informatica & Scuola [80], which is a committee of university scholars collaborating with and advising the Italian Ministry of Education on the CS and CEd school curriculum.

Open Education - Open Education is another tile in my background, as I am joining several European and Italian Open Source initiatives. Since 2018, I was Leading Teacher, i.e. the role of other teachers' mentor, for the European CodeWeek [29] promoted every year by the European Commission to enhance kids participation in computing.

In 2016 I took part in the founding process and I am currently fellow of the SOS- Scuola Open Source [145], which is an innovation center and public cooperative school promoting Open Source

5 https://en.scratch-wiki.info/wiki/Scratch2015AMS; last accessed 2023.
6 http://www.scratch2017bdx.org/en/hello-world-2/; last accessed 2023.

initiatives, shared knowledge and open design. One of the main SOS scopes is to produce Open Educational Resources (OER) such as collaborative platforms, methods, instruments and processes to establish cultural centers, program and reuse software and write open source solutions, especially in the fields of type[7] and graphic design.

Lastly, since 2019 I am a member of the Open Education Italia [114] network, whose purpose is to promote open educational initiatives and mainly the creation and sharing of OER Open Educational Resources (OER) designed by teachers for other teachers. At the moment, the network has been invited by the Ministry of Education to elaborate a proposal on the creation and use of OER to be included in the PNSD, given that they represent one of its macro areas of intervention.

3.2 THE SURVEY ON OUTREACH PROGRAMS

In this section I will provide a background on the literature on Outreach Programs in K-12 Computing Education and the results of the main study I conducted on a corpus of publications. The survey is published in the Proceedings of the International Conference on Computer Supported Education (CSEDU) 2022 [5].

The work follows the methodology of the Systematic Literature Review [76, 160] to gain a general picture, and later a classification of the papers around two different features, the audience and purpose of the Outreach Programs. Among the topics of the initiatives, I mainly addressed programming, that appears to be the most frequent in the design of the programs.

Background. In spite of their diffusion, in Computing Education OP are not considered as a specific topic as such [40], but rather part of the research on CEd and primarily on introductory programming.

The studies on how novices learn computing, and teachers teach them how to, introduce single aspects of these initiatives as, for example, their audiences and targets, or topics, or languages and tools, or educational approaches and impact. [12]. Currently, interesting overviews which delineate the diversity of the K-12

7 https://ff3300.notion.site/ff3300/Tipi-pigri-d3864e236e984fe58618716f237fccac; last accessed 2023.

Outreach Programs in Computing Education are offered by a few works [12, 99], also in the form of SLR on the initiatives [5, 11, 49, 110].

The most extensive review [92] is a broad analysis on a huge corpus of publications on introductory programming and delineates the educational trends emerging in the fifteen years until 2017.

Another SLR is also focused on introductory programming and outreach in schools from 2003 to 2017 [151]. It reports many initiatives to engage K-12 students in CS and to boost their interest and motivation in programming languages, tools and programs, both in the curriculum or in the outreach learning contexts.

A third SLR [11] is focused on CS1, a specific university course and curriculum on introductory programming, along fifty years of research on the topic. It offers a classification of the initiatives and methodologies.

Programming is a very popular topic in OP, as it motivates kids to participate in STEM disciplines [71, 92, 151, 162] and, having Computational Thinking and Creative Learning as its theoretical framework [20, 161], it can also foster students in persisting and pursuing a CS career during their transition from school to University [39]. The studies on how novices learn computing, and teachers teach them how to, introduce single aspects of these initiatives as, for example, their audiences and targets, or topics, or languages and tools, or educational approaches and impact [12].

Sometimes in literature we can find an experience of OP in national or regional initiatives, designed by governments through their educational agencies or ministries, that massively address a whole segment of school population in a country, for example primary or secondary schools [14, 47, 110]. The realization of the OP is an important action to promote CEd in schools and to increase the motivation to choose CS careers during the transition to higher education.

Often, universities and colleges establish an outreach office and a representative officer to ensure the OP quality and dissemination, and a connection to partners and stakeholders.

Even if these programs are mainly designed for students, there are also complementary Outreach Programs which target school teachers along their professional development. Whether they are novices or majors in programming, these courses seem to support educators in later motivating students in learning CS.

Related works on K-12 Computing Education OP highlight the role of the design features and process in planning and promoting the initiatives [113, 118].

The problem still difficult to face and less solved in the field of outreach is the evaluation of the learning outcomes. In fact, it is hard to assess on scientific grounds the effectiveness and impact that this approach can generate, because it is often linked to very variable methods of implementation and fruition that are very unconventional or sporadic, not replicable.

Regarding the evaluation and possible certification or validation of the outcomes, by the way, there are some interesting institutional frameworks [143] and scientific contributions that can inspire designers [39, 86, 137].

Our survey returned 12 overviews of the research on OP. These papers are mainly SLR or, at times, small surveys concerning a single audience group, both an audience and a topic, or the educational purpose and the impact of interventions. In the considered decade, the first survey [81] helped us in classifying by audience/target, school level, educational approach and students' perceptions about CS.

Two works from 2014 concern OP addressing the audience of novice and prospective students and they are both from the Australasian research area [14, 41]. In 2015 a survey [139] on learning outcomes puts the accent on the impact of these initiatives.

Another couple of papers from 2016 considers OP as a way to engage students in computing. The first one is on female students' perceptions about CS [104] and the second one on gender equity in computing [59].

In 2017 we find a very important paper on pre-college computing activities to broaden participation in CS [39]. Another 2019 SLR describes specifically a group of OP for K-12 underrepresented groups [36]. A 2020 article concerns an experiment conducted with freshmen and near-peer tutors chosen among CS students. While analyzing the impact of the outreach workshops on participants, the authors give also a SLR on the programs [68].

Research Questions. The audience of the courses appears to be the main purpose in the design of the OP. Outreach Programs identify a variety of target groups according to their age, school level, gender, social identity. Concerning this point, I came up with three research questions:

- RQ1 - Which audiences do OP address?
- RQ2 - Which audiences are less represented in the literature?
- RQ3 - Which are the purpose planned for each OP audience?

RQ1 means to express an overview of the findings in literature, and then a classification of papers based on the targets of the programs. While RQ1 appears all-encompassing and settles the state of the review, with RQ2 I define the less represented groups and, if possible, suggest a reason of their minor presence in the SLR. Finally, RQ3 aims to identify the different OP purposes, given that some initiatives are designed on a defined motivational action or expected impact on learners' attitudes towards computing.

Methodology. The survey on the OP that promote Computing Education at the K-12 school stage was conducted under the methodology of the Systematic Literature Review [77, 160].

It includes studies on both categories of learners, students and teachers, excluding incoherent results, like OP for professionals. The survey is made through the Association of Computing Machinery (ACM) Digital Library search engine, both the Full-Text Collection (FTC) and the Guide to Computing Literature (GCL). After a certain number of attempts, the following combined search phrase returned 255 results:

"outreach programs" AND "programming" AND ("K-12" OR "students" OR "teachers" OR "school")

Time range. The publication dates range from 1985 to 2021, but the trend increases significantly from 2011 on. After a screening by title, abstract and text of the publications, and after deciding which inclusion and exclusion criteria adopting, we arrived to a 239 eligible records.

So we decided to narrow the time slot and to finally consider and analyze the last decade (2011-2021), which is a total number of 144 papers coherent with the search query. The Prisma chart of the review process is illustrated in Figure 3.1.

The survey returned data on the publication trend by year in the decade, then we also classified papers in two main classes by audience/target and audience/purpose.

Audience. The audience label is attributed to each paper or initiative whose purpose refers to a group that benefits from the outreach in relation to an educational or social affinity, or even gender or cultural belonging.

We classified the total number of 144 papers as follows:

- Novice students in CEd, 60 results;
- Prospective students in CS, 33 results;
- Women, 23 results;
- Underrepresented groups, 20 results;
- Rural area groups, 4 results;
- Impaired students, 4 results.

The first group of papers (60 records) concerns OP designed for **novice** students, meaning all those learners who approach computing principles, languages and tools for the first time.

In this audience we also included the teachers actively involved in a TPD, both novices and those who already teach CS but need to update their knowledge or to upgrade it [35], because of their common intention to improve their computing competences [27].

While novices approach computing for the first time, we denominated as **prospective** students (33 records) those who intend to choose, or are selected as target, to undertake a CS career.

Another significant group of papers (23 records) refers to OP designed for **women**, to motivate girls in approaching CS and to overcome cultural misconceptions about their own role in the technology field.

The main objective of these OP is to fill the gender gap in the access to STEM, and to break cultural inequalities and stereotypes about women.

In our classification women are considered as a distinct audience, although we found their presence overlapping in other

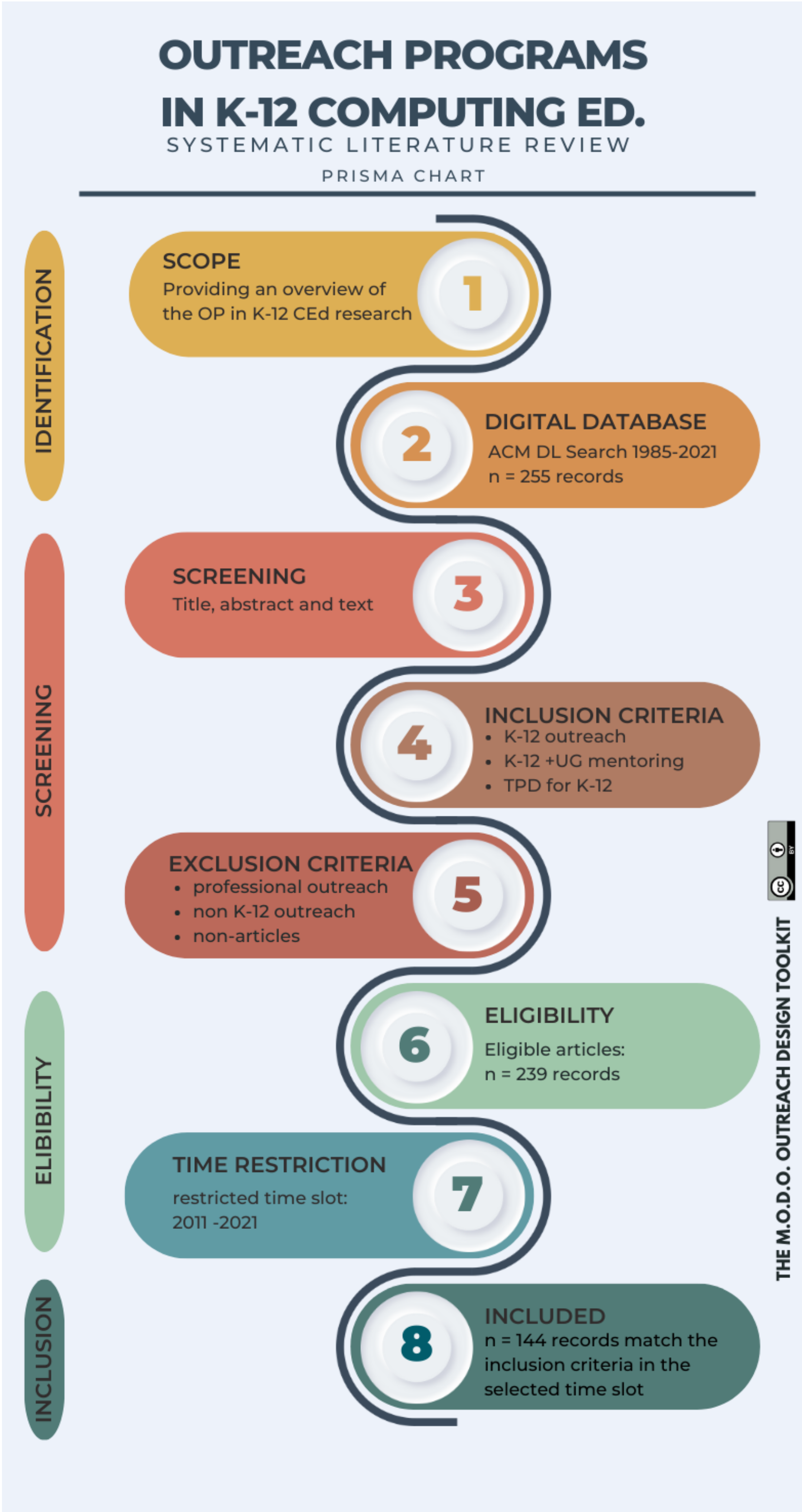


Figure 3.1: PRISMA chart of the OP in K-12 CEd SLR

programs, designed for different groups and collectively gathered with **underrepresented** students (social disadvantaged or ethnic minorities groups, 20 records). The reason is that these OP declare a common educational and social intent to guarantee equal access to CS and to remove misconceptions on computing and computer scientists.

Finally, we found a little number of records (4 each) of OP designed for **rural** area and **impaired** students.

A target label is attributed to each paper that clearly indicates a group of people who benefits from the outreach, in relation to age and education level.

The target addressed is mainly the school students population, and papers often include 2 or 3 school levels at once; however, among the total we can identify initiatives specifically and exclusively designed for primary (9 records), middle (19 records) high school (67 records).

There are OP for the TPD (14 results until 2019) and others for undergraduate students (30 results), often related to the mentoring activity they practice with middle and high school students. Five results do not permit to determine a target and 2 results are SLR.

The total number of 144 papers comprehends 34 studies that overlap on two or three different targets at once. This indicates that some programs, even if designed for different targets, maintain the same educational purpose and reshape the activities according to the school level and competences.

Purpose. The papers are classified also by the purpose of the OP. Often, it indicates also the main topic addressed by the outreach, setting aside a possible overlapping with secondary ones.

While some topics are directly CS sub-fields or regard a specific technology adopted, some others show strategies for the engagement of the participants, analyze their attitude and motivation in addressing the discipline, clarify their perceptions/misconceptions on CS.

The papers are divided by the following purposes:

- Programming, 41 records;
- Motivation, 37 records;

- Computational Thinking, 24 records;
- Perceptions, 11 records;
- Robotics, 8 records;
- Game development, 5 records;
- Unplugged coding, 3 records;
- Networks, 3 records;
- Microcontrollers, 3 records;
- Accessibility, 2 records;
- Cybersecurity, 2 records;
- High Performance Computing, 1 record;
- Not Determined, 4 records.

The Programming purpose gathers the OP on introductory or advanced programming.

The audience of the first topic, largely predictable, are mostly novices because programming represents the very initial approach to CS. This group is followed by women, as the researchers try to enhance female motivation and perseverance in CS and to promote their transition to STEM degrees with creative and collaborative activities.

Beside this group we can find substantially related OP on CT, unplugged coding and game development. A group of activities concerns physical computing, as working on robotics or with microcontrollers, or specific aspects of computing, like the accessibility or the cybersecurity issues, the networks or the High Performance Computing (HPC).

Other programs are principally designed to boost behavioral aspects of CEd and offer motivational sparks to pursue CS careers; some analyze the students' perceptions on this field and on the role of computer scientists; both these OP attempt to remove learners misconceptions, enhance learners' self-efficacy and promote positive attitudes towards the discipline.

Results. The SLR general results are shown in Table 3.1 and 3.2, and graphically summarized in Figure 3.2 and 3.3.

Further results relating to each significant audience are summarized in their respective tables and figures in the following sections.

Tables and Figures Legend: PS - Primary School, MS - Middle School, HS - High School, UG - Undergraduate, TPD - Teachers' Professional Development, ND - not defined.

Table 3.1: Audience and target of OP.

	Novice	**Prospec.**	**Women**	**Underrep.**	**Rural**	**Impaired**	Total
PS	4	-	3	1	1	-	9
MS	7	2	3	6	-	1	19
HS	25	22	11	5	2	2	67
UG	9	8	5	8	-	-	30
TPD	12	1	-	-	1	-	14
ND	3	-	1	-	-	1	5
Total	60	33	23	20	4	4	144

Table 3.2: Audience and topics of OP.

	Novice	**Prospec.**	**Women**	**Underrep.**	**Rural**	**Impaired**	Total
Programming	19	4	11	4	1	2	41
Motivation	9	11	7	9	1	-	37
CT	15	5	1	2	1	-	24
Perceptions	1	6	3	1	-	-	11
Robotics	2	3	-	1	1	1	8
Gamedev	3	2	-	-	-	-	5
Unplugged	3	-	-	-	-	-	3
Networks	2	1	-	-	-	-	3
Microcontrollers	-	-	1	2	-	-	3
Accessibility	2	-	-	-	-	-	2
Cybersecurity	1	1	-	-	-	-	2
HPC	-	-	-	1	-	-	1
ND	3	-	-	-	-	1	4
Total	60	33	23	20	4	4	144

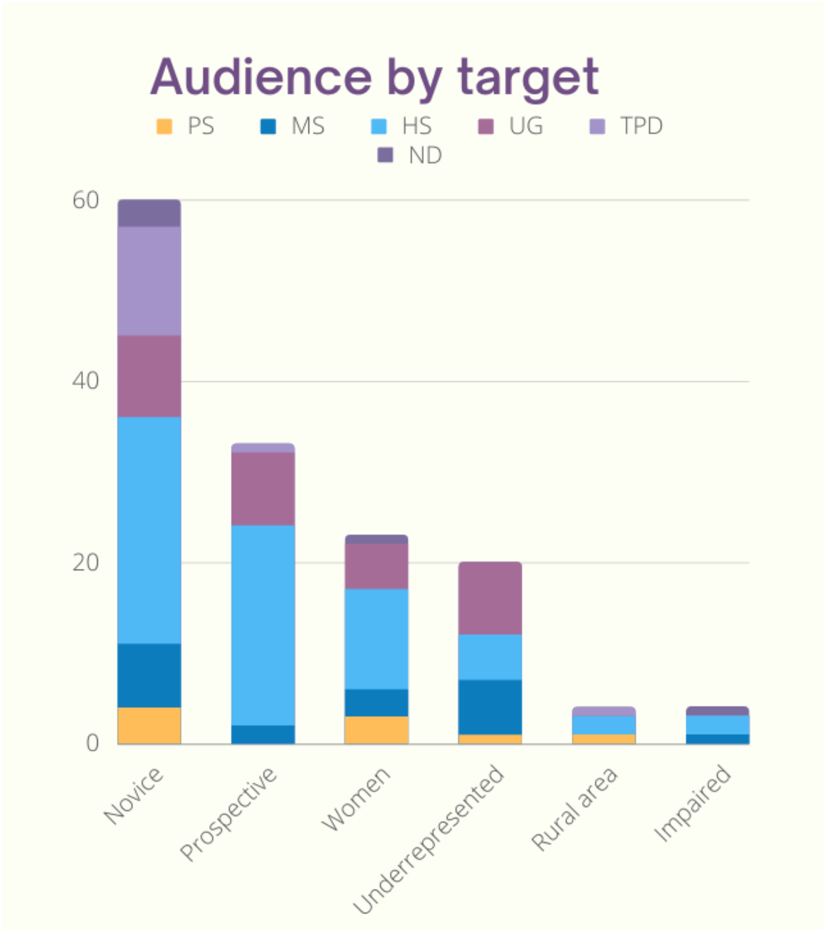


Figure 3.2: Outreach Programs audience by target

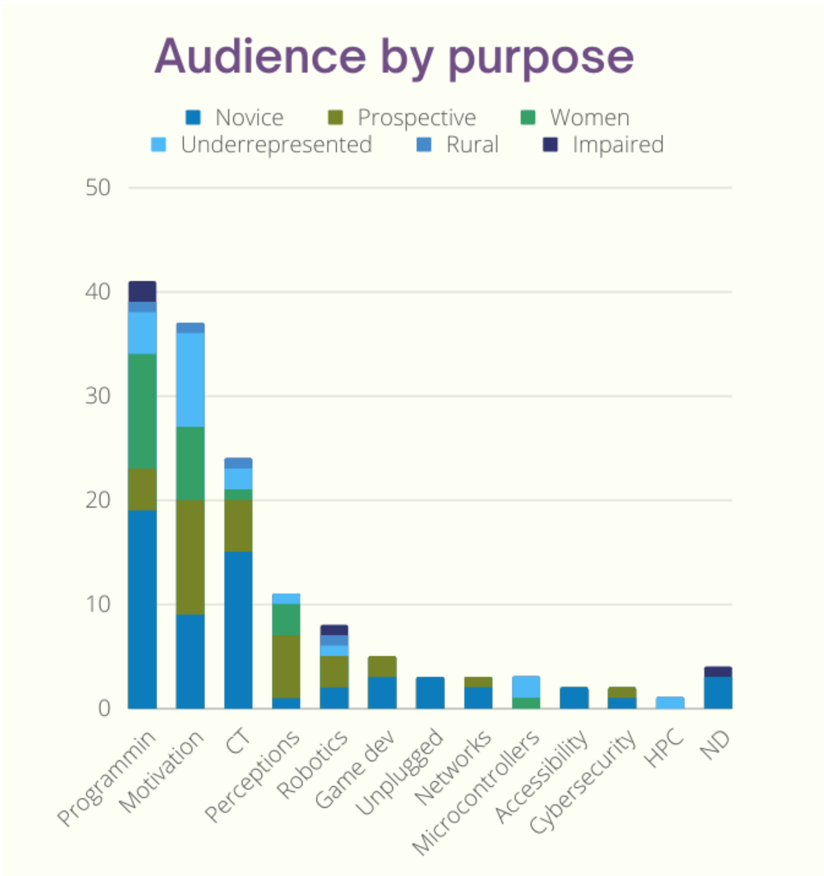


Figure 3.3: Outreach Programs audience by purpose

Women. The survey found 23 papers related to girls and women. The publications increase in 2016 when we register 6 articles. The target addressed is mostly from high school (11 records), followed by middle (3 results) and primary (3 results) schools. Other results concern 5 papers on programs for undergraduate students and 1 SLR about gender equity and female participation.

The initiatives are mainly small-scale (e.g. under 100 participants, 11 results) but there is also a certain number (8 records) of large-scale programs, for instance those addressing a whole college female target.

The most frequent topic addressed in these OP is programming (11 records). This audience group does not report the initiatives in which women are considered together with social and ethnic minorities as underrepresented groups.

These OP are mainly designed to foster women participation in Computer Science. The researchers suggest that the female motivation towards CS and STEM disciplines is more effective when raised early, e.g. addressed to a K-12 audience and when programs are offered at a young age [9, 28, 155].

However, the most recent studies underline also the existing link between the actions needed in school and in university: to guarantee the female engagement in CS, the OP involve undergraduate students in mentoring high school students [50, 101, 106]. These programs have often a positive impact, ensuring the girls' appreciation for programming and consequently their choice of a CS career during the transition from high school to college [84, 85].

Table 3.3: Women audience by target and purpose.

	PS	MS	HS	HS +UG	K1-8	K6-12 +UG	K-12	UG	ND	Tot.
programming	1	0	4	1	1	3	1	0	0	11
motivation	0	0	3	0	0	0	0	4	0	7
perception	0	0	0	0	1	0	0	1	1	3
CT	1	0	0	0	0	0	0	0	0	1
microcontrollers	0	1	0	0	0	0	0	0	0	1
TOTAL	2	1	7	1	2	3	1	5	1	23

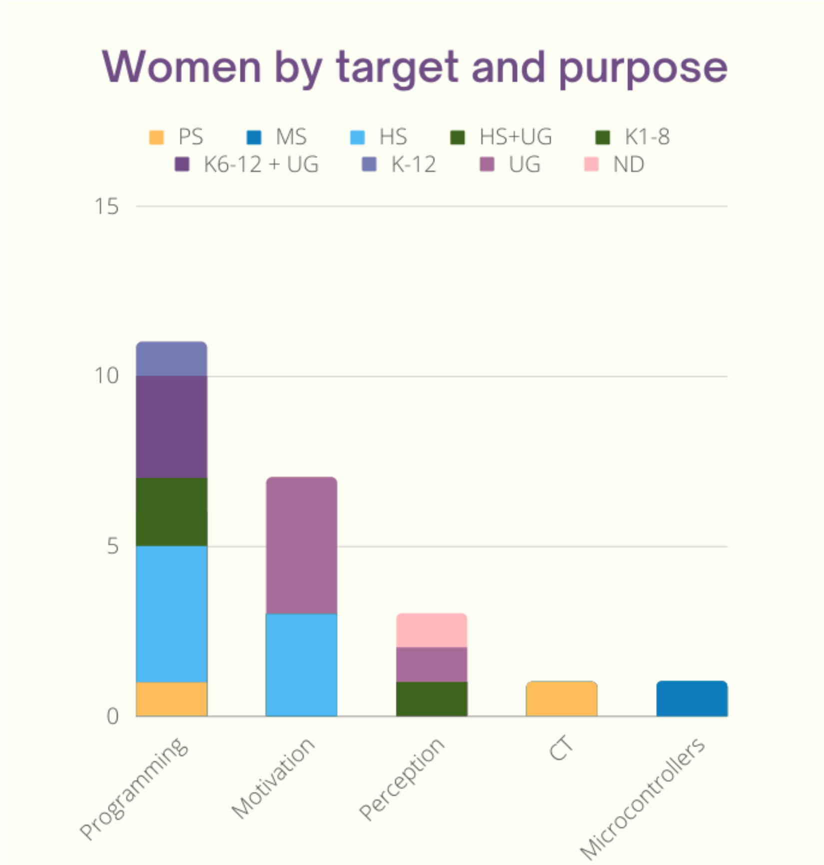


Figure 3.4: OP audience - Women by target and purpose

Some papers report studies on girls' motivation in CS and how to foster it [134], others point on the perceptions women have about STEM [84], on how to overcome both their misconceptions in CS and the cultural bias on the role and gender of the computer scientist [128]. The SLR reports the initiatives to achieve digital gender equity and to increase female participation in CS [59, 104].

Results. The results for women audience are shown in Table 3.3 and graphically summarized in Figure 3.4.

Novices. The audience of novice students in CS concerns 60 papers. The time distribution of publications is generally constant, with two peaks in 2012 (8 records) and 2014 (11 records).

The largest target addressed is the school one, with high (25 records), middle (7 records) and primary school (4 records), but there is a significant part of TPD and undergraduate initiatives (12 and 9 records each).

The outreach are small-scale programs (26 results) and large-scale (23 results), with a little number of regional OP (6 results), covering national programs in US, NZ and Europe (2014 and 2019).

The subjects addressed are mainly programming (19 records), CT (15 records), with increasing results in 2014 and motivation (9 records); a few papers on unplugged coding (3 records, until 2016 and no results from 2017 on), robotics (2 records until 2015) and game development (3 results until 2020), give place in 2021 to accessibility (2 results) and cybersecurity (1 result).

The subjects addressed, as the OP are intended for novice learners, concentrate on programming and Computational Thinking: probably, the amount of results in 2014 corresponds to the emerging global initiatives of CS divulgation as CoderDojo [31], Code.org [30], CS4all [23], European CodeWeek [29], and counting. These programs and platforms and the huge participation they got and currently get, generated a literature that describe and analyze the informal approaches to teaching and learning CS.

The presence of TPD initiatives highlights the importance of a systemic approach in designing programs. Students and teachers are key players of the same educational system, and they act in a complementary way [60, 129].

The continuing education of teachers increase the impact on pupils' awareness: students and teachers must be equally motivated to undertake CS and to get aware of the role of technology in studying and teaching [27], but also in their digital citizenship.

For this reason, besides some reviews on the state of novice programming teaching and learning, we can also find suggestions on curriculum design, educational strategies, assessment of the activities and competences acquired [25], in addition to some reflections on myths and misconceptions about CS.

Results. The results for novice audience are shown in Table 3.4 and graphically summarized in Figure 3.5.

Table 3.4: Novice audience by target and purpose.

	PS	MS	MS TPD	HS	K1-8	K6-12	K6-12 +UG	K12	UG	TPD	Tot.
programming	1	2	1	3	1	0	0	2	4	5	19
CT	1	1	1	1	1	2	0	2	1	5	15
motivation	1	0	0	3	0	0	1	1	2	1	9
game dev	0	0	0	0	0	1	0	0	1	1	3
unplugged	0	1	0	1	0	0	0	1	0	0	3
networks	0	1	0	0	0	0	0	1	0	0	2
robotics	0	0	0	0	0	1	0	1	0	0	2
accessibility	0	0	0	2	0	0	0	0	0	0	2
perceptions	0	0	0	0	0	0	0	0	1	0	1
cybersecurity	0	0	0	1	0	0	0	0	0	0	1
ND	0	0	0	0	0	1	2	0	0	0	3
TOTAL	3	5	2	11	2	5	3	8	9	12	60

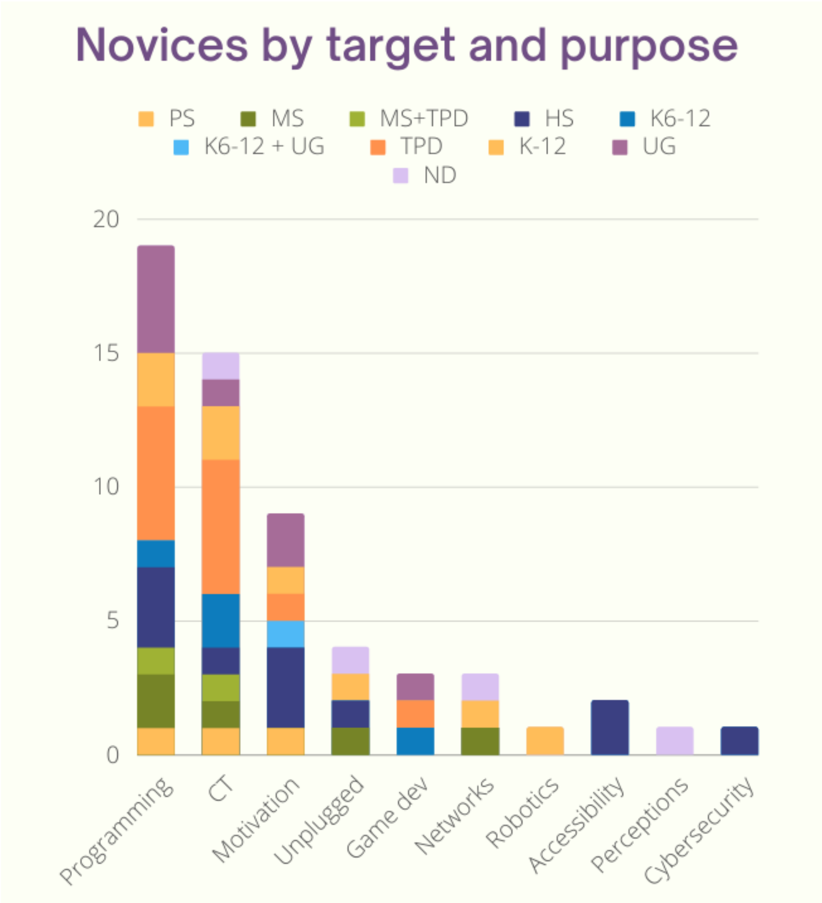


Figure 3.5: OP audience - Novices by target and purpose

Prospective. The survey regards 33 papers. The time distribution of publications registers a peak in 2011 (9 results), a progressive decreasing from 2012, a significant reduction from 2016 to 2021 and there are no results for the years 2015 and 2017. The targets addressed are mostly the high school students (20 records) and undergraduate (8 records). A few papers for middle (2 records), primary and TPD (1 record each). Small- and large scale programs equal (14 results each), while a little number of regional OP (3 results) report national massive programs in Australia and Europe in the last two years (2020 and 2021).

The purposes addressed concentrate on the students' perceptions (6 records) of what CS and who a computer scientist is

[72, 83, 147], and design motivational initiatives (11 records) to engage learners in pursuing CS careers.

Besides these topics, some OP promote programming (4 records), CT (5 records), robotics (3 records), game development (2 records), cybersecurity and networks (1 record each). The audience of prospective students in CS is considered as the target that can be motivated in computing in view of a transition from school to a CS degree. Considering the undergraduate, we can find some analysis of the motivation and the strategies to retain students in CS.

Results. The results for prospective audience are shown in Table 3.5 and graphically summarized in Figure 3.6.

Table 3.5: Prospective students by target and purpose.

	MS	HS	HS+ TPD	K1-8	K6-12	K12	UG	TPD	Tot.
motivation	0	5	2	0	1	1	2	0	11
robotics	1	0	0	0	0	0	2	0	3
perceptions	0	4	0	0	0	0	1	1	6
cybersecurity	0	1	0	0	0	0	0	0	1
gamedev	0	2	0	0	0	0	0	0	2
programming	0	0	0	1	1	1	1	0	4
CT	0	1	0	0	0	3	1	0	5
networks	0	0	0	0	0	0	1	0	1
TOTAL	1	13	2	1	2	5	8	1	33

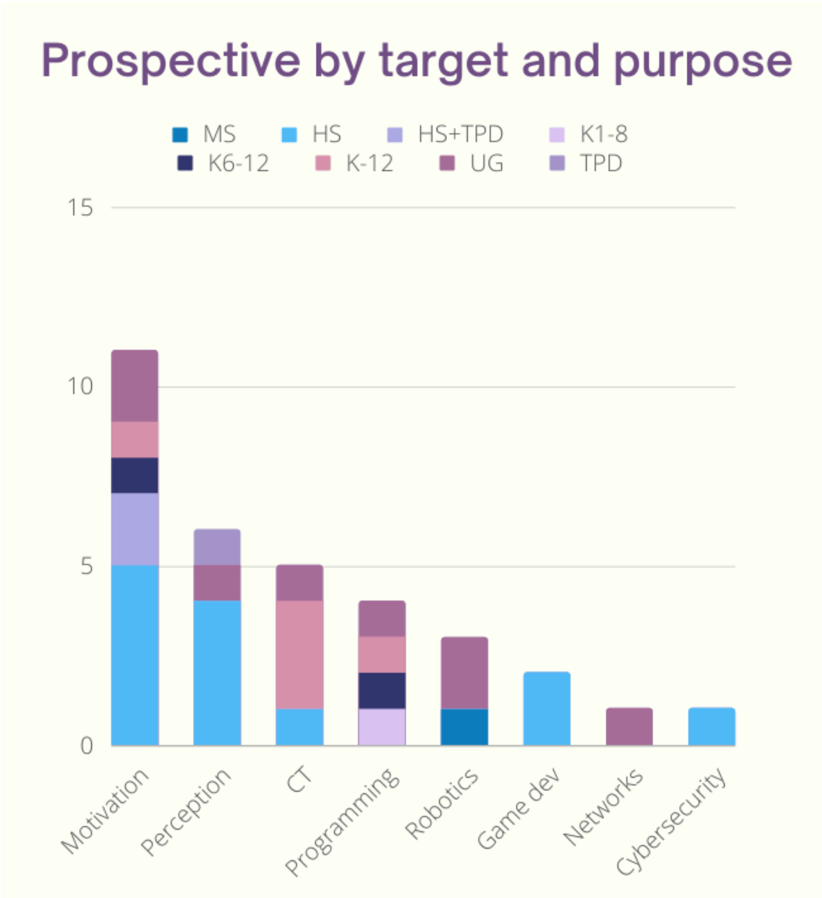


Figure 3.6: OP audience - Prospective students by target and purpose.

Underrepresented. The survey returned 20 papers. The time distribution of publications registers a peak in 2011 (6 records) and a progressive decreasing from 2013 to 2021; no results for the year 2012. The target addressed is mostly from high (8 records) and middle schools (3 records); a significant group of papers address the undergraduate population (8 records). Small- and large scale programs are the majority (7 and 10 records respectively), and just 1 record for regional OP.

The purposes addressed are mainly the students' motivation about CS (9 records) and the role of the computer scientist in our society [38] meaning to promote CS role models as a key to overcome misconceptions and remove cultural barriers. Besides these topics, we can also find programming (4 records), CT and microcontrollers (2 records each), robotics, perceptions and HPC (1 record each).

The audience of underrepresented categories in CS is vast and comprehend social and ethnic minorities, as well as women from the same groups.

This target needs specific strategies to be involved and motivated in computing [24], because they are often living at the boundaries of our society. These OP promote diversity and inclusion in CS, removing barriers to access computing education and university careers [48].

For this target we can register initiatives aiming to motivate and retain students in CS, by providing equitable access and critical thinking skills.

Results. The results for the underrepresented audience are shown in Table 3.6 and graphically summarized in Figure 3.7.

Table 3.6: Underrepresented students by target and purpose.

	PS	MS	HS	K6-12	UG	Tot.
motivation	0	0	2	1	6	9
programming	0	2	1	1	0	4
CT	1	1	0	0	0	2
microcontrollers	0	0	1	1	0	2
robotics	0	0	1	0	0	1
perceptions	0	0	0	0	1	1
HPC	0	0	0	0	1	1
TOTAL	1	3	5	3	8	20

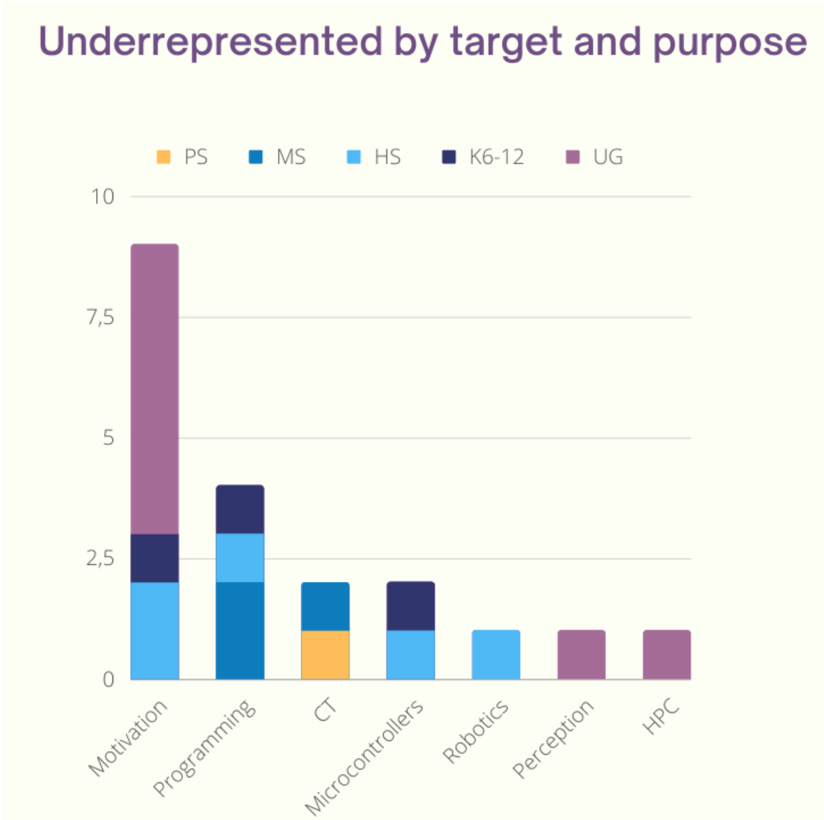


Figure 3.7: OP audience - Underrepresented students by target and purpose

Impaired. Our survey found only 4 papers on impaired audience in OP. The time distribution of publications registers only two years of publication, 2011 and 2020, with 2 records each. The target addressed is mostly from high schools (3 records); 1 result is for higher education. Small - scale programs are the absolute majority (4 records).

The subjects addressed concentrate mostly on programming (2 results) and robotics (1 result). The audience of impaired students in OP is fully occupied by visually impaired and blind people.

This target needs specific strategies and technologies to access computing, for example in robotics [90], and researchers design the programs to remove barriers to easily access CEd and university careers [142].We register initiatives that aim to motivate and retain students in CS and to guarantee an equitable access.

The choice to realize small - scale programs demonstrates how complex is to design and conduct these initiatives and, at the same time [89], to gather a class of the same target students.

Results. The results for impaired audience are shown in Table 3.7 and graphically summarized in Figure 3.8.

Table 3.7: Impaired students by target and purpose.

	HS	HS UG	K6-12	ND	Tot.
programming	1	1	0	0	2
robotics	0	0	1	0	1
ND	0	0	0	1	1
TOTAL	1	1	1	1	4

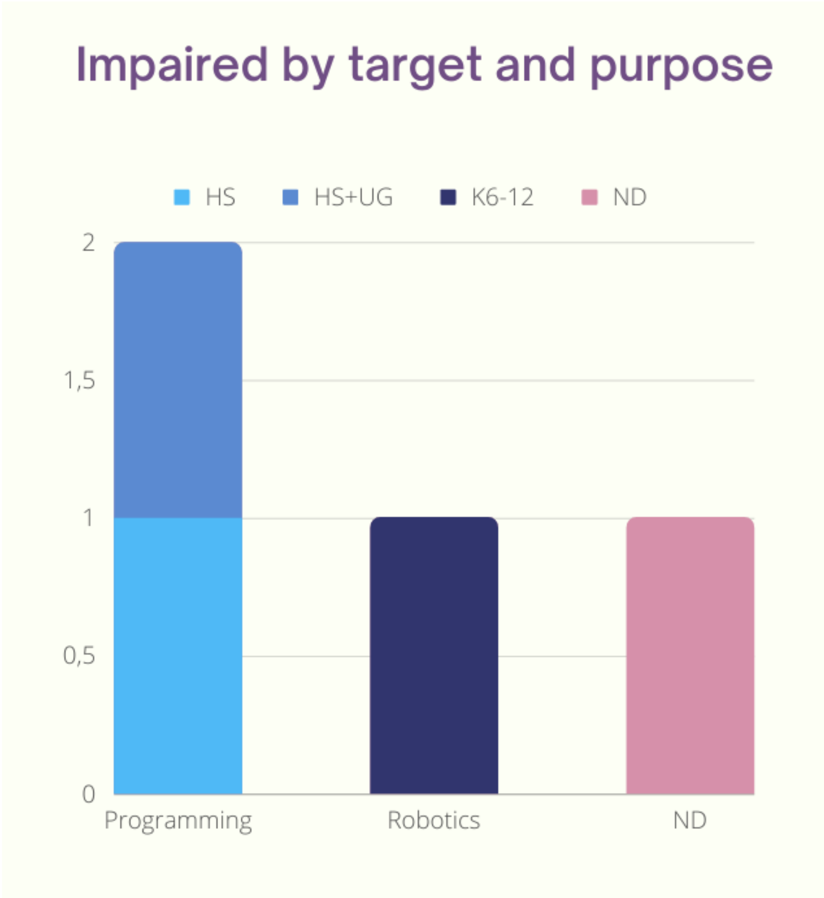


Figure 3.8: OP audience - Impaired students by target and purpose

Rural area. The OP for students that come from rural areas are a little number (4 records), generally addressed to primary and high schools (1 and 2 records respectively); 1 program regards also the TPD in the same schools. The time distribution of publications starts in 2015 and reaches 2021, but no results are found for the years 2016, 2017 and 2019. Small - scale programs are the majority (2 records).

The subjects addressed vary from robotics to programming, from CT to motivational OP (1 result each). These audiences, students and teachers, are reached by specific social strategies in order to access computing; the initiatives are designed to remove cultural barriers and to promote the choice of future careers in computing.

The choice of small scale programs demonstrates the complexity in designing these initiatives [13, 42], as students often have logistic problems like reaching the venue autonomously; researchers face issues to create a class of the same target.

Results. The results for rural area audience are shown in Table 3.8 and graphically summarized in Figure 3.9.

Table 3.8: Rural area students by target and purpose.

	PS	**HS**	**K-12**	**TPD**	**Tot.**
robotics	0	1	0	0	1
programming	0	0	1	0	1
CT	0	0	0	1	1
motivation	1	0	0	0	1
TOTAL	1	1	1	1	4

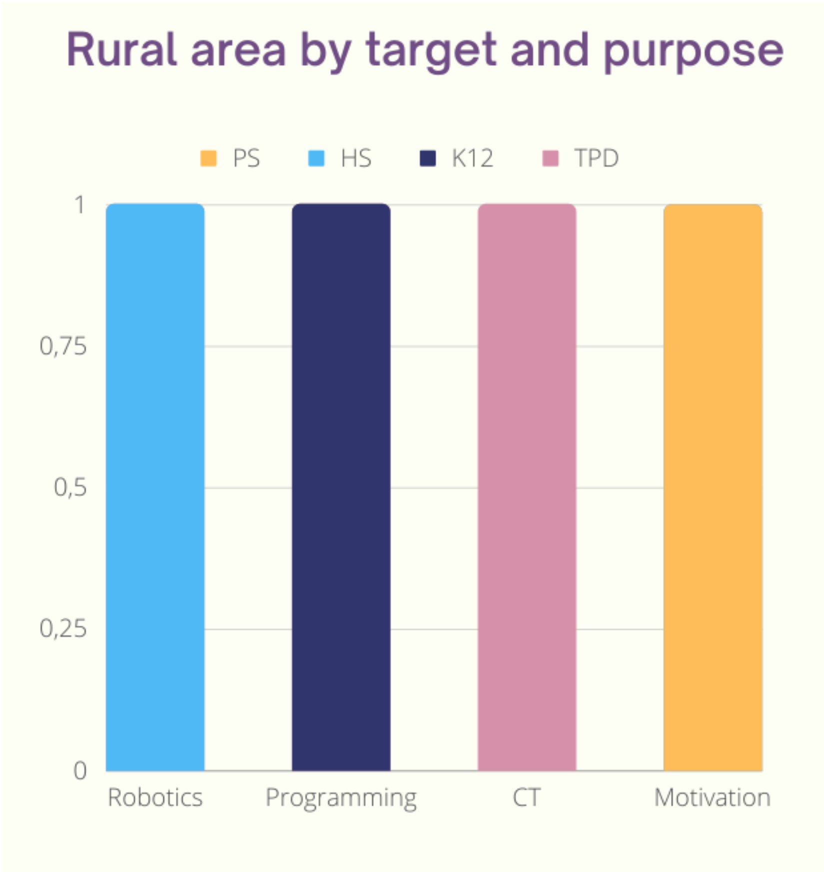


Figure 3.9: OP audience - Rural area students by target and purpose

Remarks. The survey on the OP returned a rather detailed and descriptive picture of some aspects of these training experiences designed for the K-12 range.

As for the RQ1, about which audiences are addressed by the programs the survey, although focused mainly on programming, returned a wider overview on K-12 CEd outreach and its stakeholders. It revealed a variety of audiences, among which novice students are the major group.

The designers agree that programming is an ideal ground for initiating students new to CS, and they plan educational experiences that allow them, whether they are young people or even novice teachers, to access these activities collaboratively to ensure complete satisfaction and a certain improvement of knowledge and skills.

Consequently, prospective students are the other main audience to which the activities are addressed, because designers highlight the strategic factor of ensuring the emotional involvement and the increase of motivation to guarantee participants' satisfaction but also their training success. In this very case the intention is to direct them to a possible transition to STEM undergraduate careers.

There is also a considerable presence of the female audience: these programs strongly mark the necessity to break cultural stereotypes and to boost women's participation in STEM. Proposing female role-models can impact on the girls' motivation to undertake a post-secondary path in CS and, in general, to overcome the imposter syndrome due to socio-cultural bias.

Outreach addressed to women often overlap with those intended for the underrepresented students because they have in common the same purpose: reducing the social divide of the less represented groups in the STEM fields, but also to ensure an effective participation in society and access to more satisfactory labor positions.

On the other hand, and in response to RQ2, the very few results on outreach for rural area and impaired groups mark the designers' difficulty to plan and perform these programs with these specific audiences. The sum of concerns and issues range from logistics (physical spaces easy to reach, transports, routes) to adapting programming languages and tools to the different disabilities and impairments, from infrastructures (Internet access, laboratories and venue availability) to the complexity of recruiting a significant number of participants, from accessibility to social inclusion issues.

Among purposes, which was our RQ3 for the survey, programming is the most common and often coherently overlaps with other goals chosen by the OP designers, as robotics or game development activities. These camps are relatively easy to conceive and organize also by private companies or associations and can benefit from the huge amount of platforms, languages and tools specifically intended for educational scope.

Computational thinking, chosen mainly for novices, both students and teachers, is another appealing topic that highlights how important are reasoning and solving problems while learning to program [88]. Conversely, the few results for game development, unplugged coding, accessibility and cybersecurity could suggest less interest or research commitment on these topics, except for sporadic OP addressed to novice and prospective students.

Physically impaired students (blind, deaf, Autism Spectrum Disorder (ASD)) or with learning disorders (dyslexia, dyscalculia, dysgraphia, Attention Deficit Hyperactivity Disorder (ADHD)) are often excluded from CS teaching in school because of their impairments, but they can take advantage from dedicated OP, where operators can also boost and adapt methodologies, languages, technologies and tools to their needs. As a future direction, it would be very challenging, but rewarding in terms of societal impact, to delve deeper and analyze which factors and in what measure affect the design of these experiences, in order to promote initiatives directly aimed at those less represented audiences.

3.3 THE TOOLS, LANGUAGES AND ENVIRONMENTS TAXONOMY

In this section I will present a study published at the Frontiers in Education Conference (FIE) 2022 [6] of Institute of Electrical and Electronics Engineers (IEEE), which is the design of a taxonomy of Tools, Languages and Environments (TLE) adopted in K-12 Computing Education, both in schools and outreach programs.

The research provides an analytical classification model based on an original survey of the TLE and on previous related works. It also outlines the collaborative protocol that will allow researchers to share the results of the taxonomy on a public repository.

An analysis of the most common platforms where contributors can work collaboratively is presented to show the qualitative process of identification and choice of GitHub [52] as the most reliable one.

Background. The huge development of the research on K-12 CEd discloses detailed studies on several aspects of this field, that is so crucial to social, economic and cultural progress of our society based on Information and Communication Technologies (ICT).

Teaching and learning CS in schools or outreach settings are characterized by a triad of Tools, Languages and Environments in K-12 Computing Education that increases daily.

The demand and release of new instruments specifically intended for Education and a K-12 target aims to avoid, mainly to novice students or their teachers, the complexity of using professional programming environments [97].

One of the main problems in the design of a K-12 computing class or initiative is the selection of a particular tool that meets and satisfies the choice of a specific target, school order, social group audience, programming skill level or available infrastructure. It becomes indispensable to overcome the problems of use and to increase students' experience and motivation [92].

Educational TLE represent crucial components in the design of K-12 computing initiatives. Given the variety and heterogeneity of available ones, researchers and educators face the challenge of identifying the best tool for the purpose they have set, and they invest time and effort to determine which is the best one to realize it.

In this context, a classification of this broad field becomes essential to achieve common and univocal descriptions and definitions [102]. Researchers are used to exploit their own terminology of the TLE, but it is crucial to avoid duplication of terms with synonyms and to guarantee common keywords [46] to obtain a reliable vocabulary and optimize the research strategy.

Purpose. The aim of this study is to provide teachers, but also researchers, with a categorized and shared repository of TLE employed in K-12 Computing Education programs to support the design of their courses, primarily based on the selected targets and audiences.

The purpose is to support educators in avoiding too much effort and possible failures in the choice, but also to guide a proper assessment of the initiatives.

The hierarchy of the repository items, set by keywords, has the goal of building a TLE taxonomy. It will benefit of the comparison of previous related studies on classifications and educational taxonomies.

The value of social construction of knowledge is the main purpose of our proposal. We have selected a platform which best suits collaborative work [52, 53], in order to enhance collaborators' (teachers, educators, researchers) interaction in implementing the repository.

The environment will also point to build a community where discussing and comparing the differences and issues related to the TLE terminology in order to achieve a common vocabulary.

The platform has a version control system to manage the items, which avoids errors and duplications. In this project we also provide a comparative analysis of the features of the collaborative tools that we have studied to make our final choice, which is resumed in Table 3.9.

Currently, we have already classified a first survey of TLE according to the methodology for constructing a taxonomy proposed in [102]. We are now working on an extended version of the paper, which has been selected during the FIE 2022 as a high quality research paper for the next Special Issue of the IEEE Journal Transactions on Education, entitled "Engineering Education Beyond the Pandemic" in 2023.

I selected tools already utilized in K-12 programs and recognized as effective in teaching or learning programming. To this group of TLE, I added other tools of common use both in the classroom and outreach, but not (yet) studied in the literature. Our idea is to provide teachers and scholars with a broader overview of the possible choices, according to the different learning contexts of programming and Computational Thinking.

Related works. Mainly inspired by the Taxonomy of Educational Objectives by Benjamin S. Bloom [18] and its revised version [78] as well, researchers use and adapt the classification methodology with their different approaches [21, 100, 158, 165] to develop a suitable framework for K-12 Computing Education.

The design of Computing Education initiatives often starts with the choice of a single programming language or educational tool, or from the selection of a digital learning environment that can be accessible and reliable for a specific target audience or school level.

There are many studies that introduce analytical descriptions and make an evaluation on the use or impact of a TLE taxonomy in educational settings in relation to their goals [73, 82, 103, 135]. Other papers collect information from the enacted CEd experiments and offer some wider analysis by comparing a consistent group of TLE.

They generate very interesting classifications [63, 97, 103, 125, 136], also on single fields of the discipline. The researchers list and explain their characteristics, guiding the choice of using them in the design of different programs and in various educational contexts.

However, there is not a thorough taxonomy to define the whole landscape of Tools, Languages and Environments in K-12 Computing Education, nor to level out the different terminology related to their classification.

A recent study [102] summarizes the state-of-the-art of research on taxonomies in primary and secondary Computing Education and proposes a method to generate a taxonomy by descending from other attempts traceable in the literature. I started by this accurate work, as it presents a methodology to align the taxonomies reported by a notable *corpus* of papers and suggests also a formal process to generate a coherent taxonomy.

Research questions. The literature on CEd taxonomies highlights how broad is the field of TLE and the difficulty to reach a common framework. Consequently, I will define our research according to three research questions, which we would like to investigate:

- RQ1 - What are the TLE, since the release of LOGO [144] until today?
- RQ2 - How to make a review of TLE that are not covered in (scientific) literature but commonly used in K-12 CEd?
- RQ3 - What features must a collaborative tool possess to ask the community to define a shared taxonomy of TLE?

In RQ1 I considered LOGO as the ancestor of a big part of the current TLE because of its crucial role in CEd and in CEd research, mainly for its influence on a family of languages it has successively generated [144].

The review in RQ2 is necessary to achieve a significant number of TLE, to compare and finally to adequately classify them. It is essential to include also those items not described in scientific literature but actually employed in schools or outreach settings, in spite of their lack of scientific evidence of effectiveness.

With RQ3 I aim to define the best platform for our taxonomy through a requirements' comparison of the most common tools in use. The highest quality of the collaboration features and of the accessibility means is essential to guarantee the collaborators' participation and the process of version control.

The taxonomy. The classification originates from a literature survey of the TLE in K-12 Computing Education that we made using the most common scholarly search engines. The survey was also extended to other items found through a wider web research and on the app stores for mobile devices, because they are often employed but not systematically reviewed.

We collected TLE that range from the LOGO release in 1966 until today, that we organized in a taxonomy ordered in:

- 4 classes;
- 15 categories.

The 4 **Taxonomy Classes** are:

- Languages;
- Microcontrollers, boards and robotics;
- Platforms/courses/MOOCS;
- Games and apps.

The 15 **Taxonomy Categories** are:

- TLE name, Website URL;
- Release year;
- Release source - DOI;
- Lead researcher;
- Research/development team;
- Affiliation of the team;

- City;
- Country/Countries;
- Download;
- Paradigm;
- Purpose;
- Characteristics;
- Target/audience;
- Foundation;
- Community of support.

The classes express **4 groups of TLE**, i.e.:

- programming languages with a main educational purpose;
- tangible/physical computing tools;
- platforms and environments for learning to program;
- educational games and mobile apps.

In Class 1 I have included any language designed for educational purpose and for teaching/learning programming and Computational Thinking. Some languages support also physical devices or robotics kits, some others have multidisciplinary purposes, like teaching coding for generative arts or music.

In number 2, the class of tangible or physical computing, I have collected tools like microcontrollers, single-board computers and educational robotics kits and tools.

Among platforms in Class 3, I have considered any environment where students and teachers can learn and experiment programming and CT.

Finally, I grouped under Class 4 the educational games and mobile apps for learning to program while playing.

Every TLE is also analyzed according to its general description data as well as to a qualitative analysis of its features and educational purposes. The different categories encompass data on the research and development leader and team; the website URL; the type of download (free or paid). The classification includes also the programming paradigm of the TLE; the formative objectives and programming skills attended; the specific educational purpose; the existence of a possible common ancestor to establish its membership in a family of languages.

In the taxonomy I have also identified and inserted some features related to the social aspects of programming, such as the existence of a community of support and learning, and finally whether the TLE is sponsored by any profit or not-for-profit organization (like a Foundation).

Collaborative platform. Designing and implementing in a collaborative way the K-12 Computing Education Taxonomy raise a set of non-trivial challenges. These directly derive from the need to ensure the highest reliability and quality of the collected TLE. To ease this process, we investigated existing collaborative tools and platforms to define the cooperation protocol and the presentation strategy of our taxonomy. We specifically considered the most adopted and popular online platforms for collaborative editing, visualizing/presenting, and uploading/downloading data.

In our analysis, we examined **five platform requirements**:

- *Easiness of use*: the platform should allow contributors to access and submit new records easily;
- *Automatic Presentation of the taxonomy*: the platform should enable the automatic generation of a rich presentation of the taxonomy at every newly added record;
- *Contributor Covenant*: the platform should support a code of conduct, which regulates the contributions to the TLE platform to ensure high reliability/quality of records;
- *Users Roles*: the platform should ensure different users' roles as admins, contributors, and viewers;
- *Accessibility*: the platform should provide accessibility from any device (desktop, mobile, etc.) for both editing and visualizing.

Tables 3.10 and 3.10 compare the investigated platforms and the above requirements for each of them.

Table 3.9: Comparison of collaborative platforms to contribute to the K-12 Computing Education Taxonomy.

Name	Description
Google Sheet [54]	Web application for collaborative online creation and update of spreadsheets through the Google Drive website.
Zenodo [164]	General-purpose open repository. It allows researchers to deposit research papers, data sets, research software, reports, and any other digital artifacts by accessing them using a DOI.
MediaWiki [105]	Free and open-source server-based wiki software that allows the definition of static web pages using the wiki markup language.
Notion [111]	All-in-one workspace where users can write, plan, collaborate, and get organized. Notion provides the building blocks for generating layouts and a toolkit in the form of "advanced wiki".
GitHub[52]	Provider of Internet hosting for software development and version control using Git. GitHub also offers several other features, such as GitHub pages [53] for building static websites and the definitions of actions, which are a piece of event-driven execution.

Our solution. To implement our collaborative editing and presentation process, we finally chose the GitHub platform [52] thanks to its support of contributor covenants and the possibility to personalize the presentation of data via website templates.

Further, GitHub enabled us to design our contribution protocol using versioning control and the Pull Requests (PR) mechanism. PRs allow team members to be notified of contributions to the repository so that they know they can review the proposals. In other words, GitHub provides a forum for discussing the quality and reliability of PRs before being included in the central repository.

Table 3.10: Comparison of platforms requirements for the K-12 Computing Education Taxonomy.

Name	Ease of use	Auto Presentation	Contributor Covenant	Users Roles	Accessibility
Google Sheet	*High*	*Low*	*No*	*Low*	*High*
Zenodo	*Low*	*Low*	*No*	*Low*	*Medium*
MediaWiki	*Medium*	*Medium*	*No*	*High*	*High*
Notion	*Medium*	*Medium*	*No*	*High*	*High*
GitHub	*Medium*	*High*	*Yes*	*High*	*High*

We will set up a GitHub repository providing the implementation of our collaborative contribution protocol and automatic presentation building process depicted in Figure 3.10. It consists of the following **5 features**:

- a CSV file which collects and stores the contributions to the taxonomy, in order to support versioning control;
- a code of conduct that defines standards on how to engage contributors in our community;
- a PR template, which allows contributors to submit changes to the CSV data taxonomy;
- the source code of our presentation site, written with the Jekyll [67] static site generator used by GitHub pages [53]. Our presentation site will start from the taxonomy stored in the CSV data file, and will produce an interactive and responsive website to explore, filter, and visualize them;
- a GitHub presentation site-building action, which will be triggered every time a PR is merged into the central repository and that generates an updated version of the presentation site based on the last version of the taxonomy.

As shown in Figure 3.10, our methodology comprises several initial cooperating editing steps (left side of the figure) based on versioning control of data and the PR mechanism for allowing

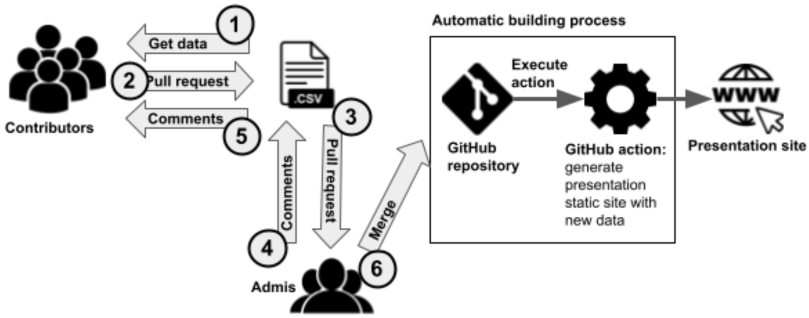


Figure 3.10: The collaborative editing process of K-12 Computing Education Taxonomy.

the revision of the contributions by the Admins community. After the editing phase, the system automatically generates a new presentation site version for each accepted contribution (PR merge) via the GitHub action.

Remarks. In this section, I presented the design of a taxonomy of the TLE adopted in K-12 Computing Education and the importance of a collaborative environment for its implementation.

In our study, we have comparatively analyzed a number of different tools designed for collaborative contribution, and finally we gave evidence of the best features that the community can benefit of, based on quality, reliability and accessibility of the tools themselves.

Therefore we can summarize the answers to our RQ:

RQ1. The panorama of the TLE since the release of LOGO is actually wide, however it is possible to gather and classify them. The *K-12 Computing Education Taxonomy* is the result of an analysis of their characteristics and it supports the definition of their value in the design and assessment of the CEd programs, in school or outreach.

RQ2. It can be hard for a single researcher to make a reliable and qualitative review of this vast catalogue. A collaborative tool, as a social platform, represents the most effective instrument for scholars and educators to participate, to share results and to amend errors. Online communities usually achieve great improvements in common knowledge and contribute to the advancement of research.

RQ3. Proper tools are essential while working collaboratively, and our previous comparative analysis suggests the most reliable platform for the presented taxonomy. The protocol needs very specific features, and we showed that each platform can really condition or instead allow contributors to participate Moreover, admins can manage the pull requests and supervise the version control protocol.

Future works. We intend to regularly collect the contributors' feedback during the collaborative participation in the taxonomy. A qualitative analysis of their comments is a starting point to monitor the community engagement and in order to make significant improvements to the platform.

We also aim to collect and link experiences from various educative contexts, a catalogue of good practices from worldwide educators, in order to provide further guidance to teachers and researchers of the community when choosing a TLE for their activities.

In conclusion, the K-12 Computing Education Taxonomy platform supports educators and scholars in identifying, defining, consulting, and implementing a catalogue of TLE for classroom or outreach activities.

It also contributes to review the TLE commonly used in Computing Education but not analyzed in the scientific literature. It provides a reliable process of shared classification and represents an accessible environment to enhance collaboration among researchers to add further findings.

According to the principles and impact of Open Science [119], through our repository and taxonomy we encourage collaboration within the community of researchers. The effort, ensured by the use of the GitHub platform and its features, adopts the system of the pull requests and the version control updates. The results of the K-12 Computing Education Taxonomy are publicly available under a Creative Common License 4.0 Share-alike.

3.4 CO-DESIGN AND COLLABORATIVE LEARNING IN PRACTICE

In this section I will present a group of papers that we published, together with a team from our research ISISLab at the Department of Computer Science of the University of Salerno, about an original Digital Storytelling (DS) environment called *Novelette*.

We designed the editing tool and the educational platform and co-designed many features with the teachers first (in a controlled setting) and later also the usability with the students (in a real classroom context).

We presented the environment for the first time at the International Conference Information Visualization (IV) 2020 [2] and discussed the results of the experiments on the engagement of students at International Conference on Methodologies and Intelligent Systems for Technology-Enhanced Learning (MIS4TEL) 2021 [4]. Last, we published a third paper in the IEEE Journal Education Society where we show the results of the co-design process on the usability of the platform [3].

The *Novelette* experience returned us useful prompts for the design of an OP, respectively on two crucial aspects: the engagement of the audience in the co-design of the session, and the research and improvement of the usability of a tool, platform, and learning environment.

Background. Storytelling represents the skill of inventing and authoring stories, behaving as a simple but powerful method to share experiences and convey knowledge [154].

It empowers learning practices, and enables knowledge sharing and communication, critical thinking, and technical skills development [107]. Digital Storytelling is a practice in which users are supported by technological solutions [26], such as authoring interfaces or digital learning environments. As a result, besides authoring stories by traditional means, such as paper and pencils, users can experience visual storytelling by relying on digital media, including images and text [26].

Educational DS has a robust tradition that relies on Jerome Bruner's research concerning the role of narratives as an opportunity for learners to share knowledge [22].

It is perceived as a powerful technology-enhanced learning approach [163], widely adopted at each education level, from primary to secondary [163].

Purpose. To mitigate the lack of tools to support learners in improving their creativity through DS, we proposed *Novelette* [2, 4], a free and open-source digital learning environment to invent and author visual stories.

As a DS authoring interface, *Novelette* supports learners in creating, refining, and rendering stories containing textual and graphical components.

As its main novelty, it embeds narrative artifices proposed by Gianni Rodari [130] to scaffold pupils in inventing their stories. In particular, *Novelette* embeds the suggestion mechanism to guide students in playing on words by iteratively exploring synonyms and analogies and enables them to continue someone else's story through the *incipit* mechanism.

To identify learners' needs and take care of learning requirements, *Novelette* results from a collaborative design approach in which educators have been actively involved in proposing features and revising the final prototype. As a result, *Novelette* has been designed not only *for* education, but also *with* educators. As it aims to support educators and learners in performing educational DS, collecting opinions from both target groups is crucial.

Among the proposed narrative artifices, we focused on the suggestion and the *incipit* mechanisms as they represent some of the best-known approaches to invent stories [3]. I will provide an introduction on both.

Suggestion. The suggestion mechanism represents any technique to be inspired by and to overcome the blank page syndrome. To introduce this literary artifice, Rodari uses the metaphor of "*throwing a stone in the pond*": when the stone touches the surface of the water, many concentric waves start.

The human brain behaves in the same way when a word comes to mind by recalling mental images, associations, metaphors, personal experiences, and feelings. Rodari suggests thinking about a small set of words, a single word in the simplest case, and exploring synonyms and analogies by playing on terms. The process generates a chain of words that lead narrators to recall images that may inspire stories.

Incipit. The *incipit* mechanism represents the possibility of continuing someone else's story. Gianni Rodari uses the story of Pinocchio to introduce this concept by challenging readers to think about "*what happened to Pinocchio when he finally became a* *child?*". While the Pinocchio tale ends with the transformation of Pinocchio into a real child, Rodari uses it as a starting point for another story.

In this way, basic narrative components, such as characters, time, and place, may take new paths and lead to a surprising or unexpected development. According to these literary artifices, students may be provided with an incomplete story to continue through their creativity.

Related works. Several DS platforms have been proposed to create media or data stories. Although journalists and media curators widely use these tools for professional scope, they cannot be easily adopted as educational DS platforms because they lack features considered crucial in learning settings, such as: support at the class level, group management, and literary support as defined by Rodari.

Table 3.11 provides an overview of the related work supporting educational DS, implementing group management features, and/or literary artifices as defined by Rodari to support the story invention phase. Tools are classified as *Digital Storytelling editors* if they are provided with story authoring and publishing mechanisms. *Class management* feature represents the support at the class level, group management, and supervision from the educator's perspective. *Incipit* and *Suggestion* represent any approach to embed Rodari-style artifices in the DS platform.

Novelette. *Novelette* is a digital learning environment that supports educators and learners in inventing and authoring linear stories.

It provides target users with an editor interface to perform Digital Storytelling and features to support learners in inventing stories by thinking out of the box and improving their creativity [2].

The inventing step is enabled by embedding the *incipit* and the suggestion mechanisms in *Novelette*.

Novelette is based on *student* and *teacher* roles. While learners play the storyteller role as they can invent and tell any story of interest, teachers can organise classes, supervise created stories, and publish them.

Table 3.11: *Novelette* - Digital Storytelling tools comparison.

Tool	Year	Digital Storytelling editor	Class mgnt.	Incipit	Suggestion
Fabula	2017	-	-	✓	✓
Communics	2020	✓	-	-	✓
Wakelet	2019	✓	✓	-	-
UTellStory	2020	✓	✓	-	-
StoryJumper	2020	✓	✓	-	-
Storyboard That	2020	✓	✓	-	-
Pixton	2020	✓	✓	-	-
Storybird	2020	✓	✓	-	-
Comic Life	2020	✓	✓	-	-
Novelette	2020	✓	✓	✓	✓

Class Management Features. Users who play the teacher's role are provided with a managerial dashboard (Figure 3.11) which enables the class management features summarised in Fig. 3.12.

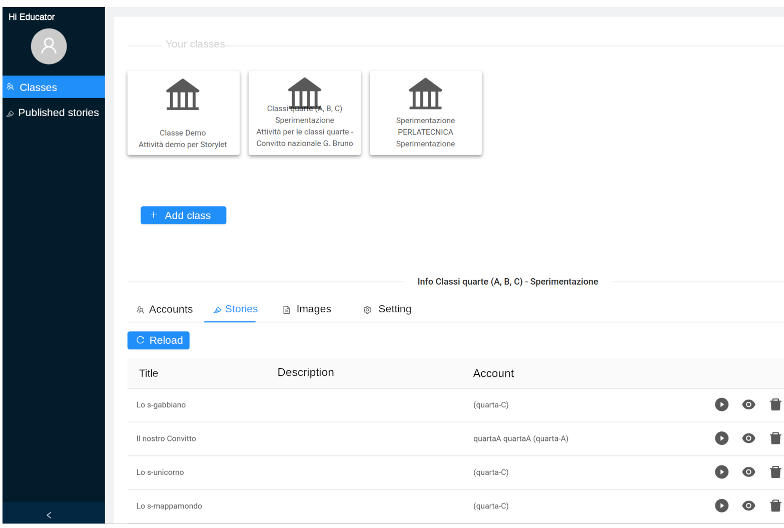


Figure 3.11: *Novelette* - Educators' managerial dashboard.

The *Incipit* mechanism. Educators may provide learners with *incipit*s as templates. Teachers can introduce a topic by providing

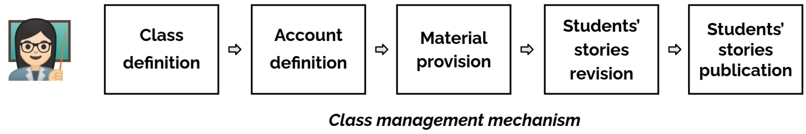


Figure 3.12: *Novelette* - Class management workflow.

learners with the beginning of a story, and learners are encouraged to continue the story starting from this premise.

Alternatively, teachers can bind the story initiation and its ending, and they can ask learners to interpose their stories by satisfying these constraints. For instance, it can be used to narrate different versions of the same story where there are two or more events that everyone knows, and learners have a different point of view of what happened in the meanwhile.

Figure 3.13 shows the *Novelette* creator interface when learners start with an educator *incipit*.

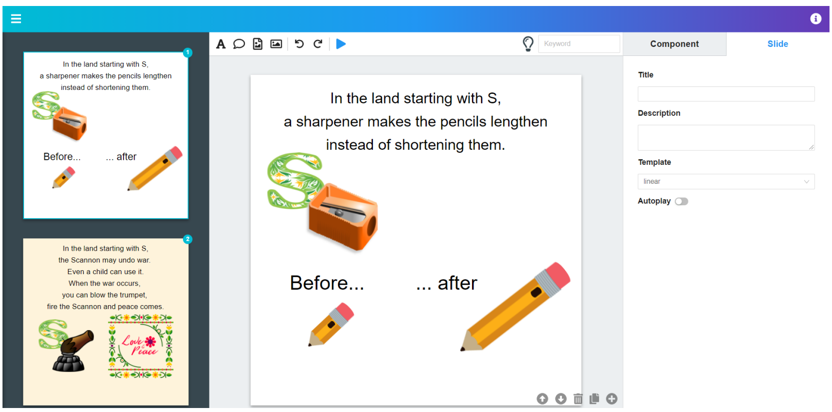


Figure 3.13: *Novelette* - The creator interface.

Digital storytelling editor. Users who play the student role can author stories according to the workflow described in Figure 3.14.

Once authenticated, learners have to select one of the available templates. Some of them are default templates proposed by the platform, for example the empty template, while others represent the educators' defined *incipits*. Once the starting point is chosen, learners access the *Novelette* creator component, that is,

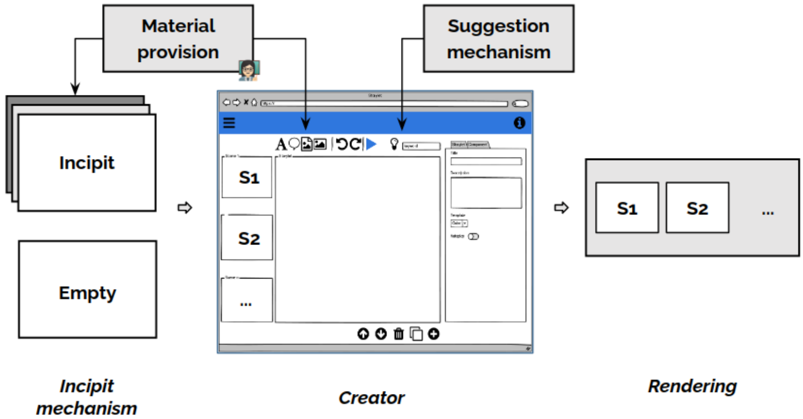


Figure 3.14: *Novelette* - The workflow to create stories, student side.

the creator interface to invent and author stories, whose interface is visible in Figure 3.13.

Suggestion mechanism. If users experience blank page syndrome, *Novelette* offers a suggestion mechanism of analogies, synonyms, and rhymes based on a user-defined word. Starting from a word of interest, *Novelette* groups synonyms by senses, as in Figure 3.15.

Figure 3.15: *Novelette* - The suggestion mechanism.

Novelette has been designed and developed by combining user-centered and participatory design to address educators' needs and actively involve them in proposing and reviewing supported features [3]. Developers actively involved interested educators in testing the platform, verifying in first person the ease of use, and suggesting either integration or modifications to supported features [3].

The main goal of this evaluation is the assessment of the *Novelette* usability for both educators and students. We performed a two-round assessment.

First, we tested *Novelette* by involving educators participating in the design approach and performing the assessment in a controlled environment. This educators' evaluation round verifies whether and to what extent *Novelette* satisfies end-users expectations maturated during the design approach.

Second, we evaluated the usability of *Novelette* in real settings by involving three primary school classes that used *Novelette* at school, without being supported by developers.

Remarks. *Novelette* has been designed not only *for* education but *with* educators as they have been actively involved in the design approach addressing concrete needs that they face daily with pupils.

As a result, one of the articles assesses the usability score by involving both educators and learners to verify to what extent *Novelette* is perceived as a usable visual storytelling digital environment, as well as to develop students' creativity [3].

Novelette is considered *usable* by educators and learners as it achieved an SUS score of at least 75. Qualitatively speaking, involved participants successfully created stories stating that the interface is easy to be used both according to educators and children.

According to involved educators, it is easy to learn how to exploit *Novelette*, stressing the intuitiveness of the proposed tool. Moreover, it demonstrates that users do not require technical support to use *Novelette*.

The same pattern can be observed by working with students. They rarely asked for any support and, in most of the cases, peer support was enough to overcome challenges.

By discussing with the moderators, children loved working collaboratively, and they asked for peer support to invent and author stories together. In fact, they think about the inventing part and the characters together to share and exchange opinions with friends.

Novelette has not only to be designed *for* educators, but it resulted by a collaborative design approach to propose a tool that actively involve teachers in the entire design and development phases. Thus, *Novelette* has been designed *with* educators.

Part IV

THE DESIGN

Simplicity is about subtracting the obvious, and adding the meaningful.

John Maeda, *The Laws of Simplicity*

4 THE DESIGN OF OUTREACH PROGRAMS

Outreach Programs promote a continuous experimentation of tools, environments, scopes and methodologies [12].

The approach to learning is strongly encouraged, fundamentally intentional and based on personal motivations, whether it is conscious or not [99]. Researchers and educators deal with the real motivation of participants towards computing and technology, so they cope with emerging formative needs and also to guarantee a proper organization and a flexible setting to perform the programs [20].

Due to the unconventional nature of the initiatives, Outreach Programs have a peculiar design and can disclose few **concerns** to researchers, educators or extracurricular volunteers. These educational environments encourage designers to find new solutions to distinctive problems or to engage particular target groups or audiences.

Moreover, they are often integrated into the life experience of learners. This concern connotes some sessions under a social and somehow political perspective, because it fits into a more general discourse on social inclusion and recovery of the digital divide.

Given these premises, it is necessary to carefully analyse the **purpose** of the initiatives in relation to the addressed **targets** or **audiences**, the educational **approaches** and the available **tools**, but also to describe them on the basis of an adequate **design process**.

The design of K-12 Computing Education OP takes different shapes depending on the context in which it occurs. Planning, identifying problems and their solutions, evaluating the outcomes, they all represent aspects of a discipline that supports the implementation of educational programs.

In this chapter I will provide a taxonomy of some design guidelines of the outreach and out-of-school Computing Education programs.

4.1 RESEARCH QUESTIONS AND METHOD

Since OP have profoundly different features and purposes, the design process is not standardized and cannot be applied indistinctly to every initiative. For these reasons, our research focuses on the design concerns that K-12 Computing Education programs can occur to OP designers during the ideation, or even happen to them after the implementation of a session.

Researchers, educators or practitioners who conceive and plan these pathways do face multiple challenges and devise a few solutions. Sometimes instead, they see the encountered difficulties only after the realization of the program, during the follow-up phase.

Given that it interests and involves learners, even in groups or communities, CEd design and its researchers must necessarily also use social science methods [152]. The outreach goals require instructional design skills and an inter- and cross-disciplinary approach to the research in Pedagogy and Learning Sciences in addition to that in Computer Science. In Chapter 3 we saw that designers often intend to meet the learners' needs and promote their full realization as individuals within a community, or as citizens within society.

Despite the differences between programs, however, there are also similarities that can lead to sketch the Outreach Programs design guidelines and stages.

Based on the literature evidence as well as on my personal insights from pedagogy and social sciences, I elaborated the following research questions in order to model the OP design process:

- **RQ1** - What are the OP design process categories and features?
- **RQ2** - What are the OP design process steps?

The planning of educational paths grounds on a few pillars that guide designers to the choice of an educational or social feature, a type of pedagogical approach, the selection of a target and the possible areas of personal development to be increased.

In response to RQ1 I adopted a taxonomy which incorporate the analysis and classification of the literature. What emerged in response to the question is a cross-cutting taxonomy in which the **three design pillars (Principles, Aims, Strategies)** I have delineated are splitted into subcategories:

1. **Principles**
    - Skills
    - Self
    - Interaction
    - Citizenship
    - Openness
2. **Aims**
    - Skills - Increase creativity, critical thinking, positive attitude, persistence, performance, self-regulation; promote STEM disciplines; boost TPD;
    - Self - Fill the social and gender gaps; promote inclusion, equity, engagement, empowerment, self-efficacy and regulation; adopt accessibility and Universal Design for Learning; overcome underrepresentation, bias, imposter syndrome, cognitive load, procrastination;
    - Interaction - Promote social interaction and collaborative learning; increase collaboration, team building;
    - Citizenship - Promote digital awareness and the safe use of the Internet;
    - Openness - Promote generalization of the programs, scale, reproducibility; adopt iteration, reuse of resources and practices, accessibility.
3. **Strategies**
    - Pedagogy;
    - TLE;
    - Mentoring;
    - Team/Community building;
    - Artifacts;
    - Co-design/customization;

- Role-models;
- Evaluation/lessons learned;
- Dissemination;
- Duration.

On one hand, I attempted to define some general and more theoretical design concepts to exploit during the planning; on the other instead, there are practical requirements to consider and that set a concrete group of items. Some categories set real constraints, somehow impossible to overcome; some are just flexible facets that can be adapted to the goal or targets addressed.

4.2 THE DESIGN PILLARS

The planning of educational paths grounds on a few principles that guide designers to the choice of an educational or social feature, a type of pedagogical approach, the selection of a target and the possible areas of personal development to be increased.

These principles are of a general or even more specific order; they pertain to functional problems, meaning they are linked to specific demands of the learners and their approach to knowledge, or are ethical-social issues, deriving from the dynamics among students in the fellowship with peers, the relationship with the family, the school setting and the effects of the social context in which they live.

From the analysis of the literature, I finally highlighted these **Principles**:

- Skills;
- Self;
- Interaction;
- Citizenship;
- Openness.

With respect to the aforementioned principles, the main objectives deriving from them concern their subsequent development.

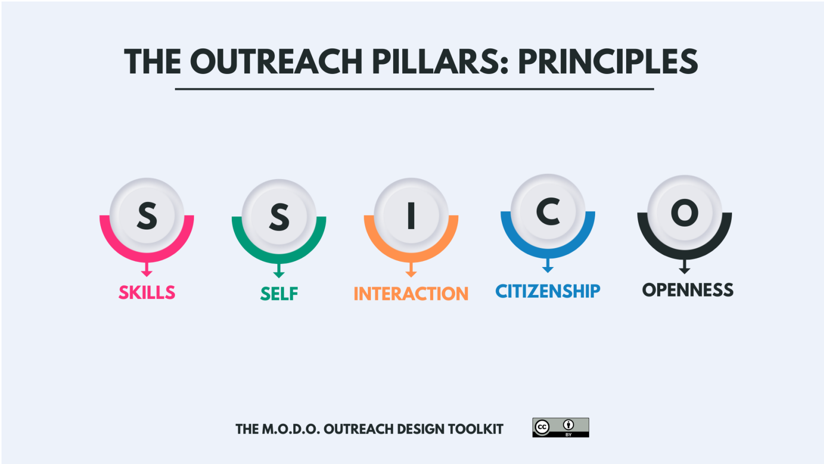


Figure 4.1: The Outreach Design Principles.

In addition, some other educational aims emerge, that the designers adopt and pursue in the planning of the activities. Some are specific for peculiar targets or audiences and to solve the main problems associated with their groups, but there are also purposes of a more general nature, which satisfy the same need even for different targets. The principles are resumed in Figure 4.1.

The **Aims** for each above principle are:

- Skills - Increase creativity; critical thinking; positive attitude; persistence; performance; self-regulation; promote STEM disciplines; boost TPD;
- Self - Fill the social and gender gaps; promote inclusion, equity, engagement, empowerment, self-efficacy; adopt accessibility and Universal Design for Learning; overcome underrepresentation, bias, imposter syndrome, cognitive load, procrastination;
- Interaction - Promote social interaction and collaborative learning; increase collaboration, team building;
- Citizenship - Promote digital awareness and the safe use of the Internet;

- Openness - Promote generalization of the programs, scale, reproducibility; adopt iteration, reuse of resources and practices, accessibility.

Outreach principles and aims shape the scaffolding of the design method, but are implemented by didactic and organizational strategies.

These approaches affect both the planning and implementation of the program, but also all aspects related to the well-being of the participants, their full personal fulfilment, the social interaction dynamics that support the learning process.

Among **Strategies** we find:

- Pedagogy;
- TLE;
- Mentoring;
- Team/Community building;
- Artifacts;
- Co-design/customization;
- Role-models;
- Evaluation/lessons learned;
- Dissemination;
- Duration.

4.2.1 *Principles*

At the foundation of outreach in Computing Education there are general pedagogy principles, but also social and cultural components of everyday life.

These principles pertain to different forms of learning and certainly converge towards the goal of promoting Computer Science detached from school practice and teaching under a binding curriculum.

For this reason, this approach has the main consequence to involve those students who are generally dropped-out, intolerant or demotivated at school. In their specific case, the less conventional teaching and learning, the unstructured environment and the absence of hierarchies and evaluation have a positive impact on the well-being and integration of all the participants.

These environments also favor and guarantee the access of people regardless of their cognitive difficulties, or disabilities, or different educational needs.

In ensuring equal access and inclusion, they push researchers and educators to come up with ever new and creative solutions to ensure that everyone can participate.

Skills. Many Outreach Programs are based on the development of new or enhanced skills.

This is a key tenet in Computing Education, which is not only about computational thinking and skills, but also some skills in logical and creative thinking, social interaction for collaboration, individual ability to persistence on goals and positive attitude towards novelty, change or difficulties.

Self. The design of the programs takes into great consideration individuals, their belonging to social, ethnic and gender groups, the equal access, the drive to improve and guarantee themselves a professional future that enhances their economic status, the difficulties they may encounter at the cognitive level or in relation to their performance in activities.

Several problems are much more frequent in some social groups than in others, for example the gender gap is particularly evident in the case of women, or accessibility concerns above all the impaired students.

Interaction. A key theme in outreach design is social interaction. It is expressed on several levels, for example in the sociability of individuals when it comes to collaborating for the success of a project. Many programs have among their goals the facilitation of interaction between participants and collaborative learning; many activities are built thinking about how to facilitate the dynamics of building the working group, also in the future perspective for professional integration.

Citizenship. In our extremely interconnected world, characterized by the exponential development of some disruptive technologies, an individual growth factor is represented by the awareness towards tools, technologies and programming languages, as well as the knowledge of the ways in which algorithms regulate our online lives.

Educating kids on how to use but above all how to design software, a platform or a video game can be the starting point for developing awareness on what we do and on the consequences it has on our and others' life.

Openness. Outreach Programs often have the characteristic of being isolated initiatives, which are planned for a single session or for a short cycle of a few days.

Designers often wonder if those activities have had repercussions on learners and how to objectively measure the results, to share the results with the scientific community.

In many cases the problem of generalization and scalability arises, i.e. the possibility of exploiting the flexibility of some elements of that program to adapt it to another target, for example with a different age range and therefore with other educational needs, or for a different audience, with different group characteristics.

Another important element is the showcase of the final artifacts, the dissemination and reuse of resources, practices, educational experiences, the dissemination of activities and, when possible, of the results.

4.2.2 *Aims*

There are many purposes that can be attributed to outreach activities. Among the main ones, in many contexts it is often practiced and promoted the education for an active and aware online citizenship.

Knowing how to use the Internet correctly to acquire news and learning how to tell the proper information from fake news; knowing how to use social networks paying attention to the tricks they can hide; learning how to access institutional websites for one's own benefit; are some of the possible purposes that Outreach Programs can offer.

Women tend not to choose STEM disciplines due to cultural and familiar stereotypes. In the last decades, many initiatives were directed towards enhancing their participation in Computing, especially with sessions on programming and web development, for instance.

The shortage of women in the IT sector is now considered seriously worrying because it causes negative consequences both in terms of job placement and of professional qualifications, but also of digital citizenship and equal access to public services, civic consultations, participation to the political and social life.

A further, non-trivial consequence is also the difference in women's wages compared to those of men, which, especially in this sector where the professions are so in demand, represents a huge economic gap and factor of exclusion.

Outreach can also be a significant opportunity to remove students' cultural, social and economic gaps. For example, some concern gender belonging or coming from ethnic groups living in marginal areas.

Some categories of participants are chosen precisely for these reasons, with the aim of enriching the highly deprived social context, combating and removing prejudices on gender and belonging, introducing IT professions to those groups that otherwise would not have had the same opportunities.

Sometimes these students suffer only the consequences of their geographical location and the lack of infrastructures, such as adequate Internet connections or transports to reach school and educational centers.

Critical thinking. This principle has a very strong transversal character and ranges between several disciplines, including Computing Education.

Cognitive psychology is prompting many educators to lead students in the active practice of critical thinking rather than just memorizing large amounts of information or procedures.

Critical thinking in K-12 Computing Education supports analysis, observation and reflection on findings; it helps students to express grounded opinions about both the theory and the related practice.

It exploits the principles of clarity, accuracy, precision and evidence, leading the individual to question his own beliefs by comparing them with those of others.

It is based on observation and experience, but it implies a reworking process that guides reasoning.

Empowerment. The achievement of this objective is a direct consequence of the inclusion of empowerment as a principle in the design of the outreach.

Learners can develop their awareness to arrange their actions. Women particularly benefit from this purpose because they are often subject to numerous gender stereotypes that prevent or limit their full participation in STEM activities. The result is a lack of development in knowledge advancement and job maturation. If they are not yet oriented towards STEM university careers, outreach can boost their choices.

Underrepresented targets who come from rural or economically and socially disadvantaged areas also benefit from OP, too, because they overcome the social exclusion, digital and economic divides, family conditioning and school dropping out.

Social inclusion. There are many reasons for carrying out Outreach Programs that mainly aim to promote the social inclusion of subjects who experience a strong disadvantage on an economic and social basis.

By participating, in fact, these students can access a world of new possibilities that are often impeded to them due to the lack of tools or Internet connection, few or scarce learning opportunities on specific CS topics, or the burden that family and cultural stereotypes have on the affirmation of individuals.

Some programs also consider the inclusion of subjects who live on the edge of big towns or in rural contexts, where training opportunities are very limited and where the geographical location affects participation.

We also talk about inclusion in the case of programs for people with disabilities or minorities, when they have often the secondary objective of job placement or school advancement.

Self-efficacy. This goal is about the ability to believe in one's own ability to achieve an educational goal.

Self-efficacy strongly influences how a person can meet a challenge with their skills or making appropriate choices.

The consequence is the promotion of personal well-being, the ability to face challenges and to recover from failures.

Positive attitude. Some outreach targets and audiences, who live in contexts in which they do not have the opportunity to constantly participate in Computing Education activities, or who are influenced by cultural stereotypes, are also very slow to develop a positive attitude towards STEM disciplines.

Sometimes they convince themselves that those are subjects that cannot be tackled by everyone, and that scientists are people of uncommon intelligence. They imagine them stereotyped, like nerds or similar.

Activities that break these beliefs also help to promote greater openness to these disciplines, involve categories of students usually at the margin of society or school dropouts.

Collaboration. I have included this objective also among the principles, since in fact in some programs it is considered a goal to be achieved: didactic strategies are elaborated to promote the collaboration of individuals within the group.

From the point of view of the results, the community of students that is formed around a path can represent a collaboration tool even if it intervenes in parallel or after the realization of the activity.

Gaps filling. In K-12 Computing Education, the main gap to fill is represented by the involvement of girls in STEM subjects.

It suffers of the many familiar and cultural stereotypes that weigh on women in general, but on the sector of Information Technology in particular.

The main objective of these Outreach Programs becomes the reduction of those prejudices that girls often feed themselves, without realizing that there are no reasons for their exclusion and above all that there are subjects that can guarantee them great satisfaction but also an adequate professional and economic future.

STEM choice. Many programs aim to encourage the participation of students approaching STEM for the first time, the so-called novices.

Generally, these outreach are addressed to younger kids or groups who do not have access to Computer Science contents (or related) in the school curriculum.

The main feature is therefore that of the simplicity in the approach and the gradualness in the acquisition of knowledge and skills.

The ultimate goal is to foster a positive attitude throughout the pipeline up to the choice of the university career.

These programs also include those for the target of prospective students, usually frequenting high schools, who already have in mind or who will choose a STEM path.

Awareness. A key principle in disseminating Computing Education is awareness. It involves many areas of individual development and the relationship with technologies.

For example, it concerns the ability to select and process the information received, adopting all filtering strategies to understand its importance but also to distinguish any risks.

This principle includes activities to raise awareness on the value of technology as a means of accessing, processing and creating knowledge.

It is an enabling factor because it allows students to get to know and understand the reality that surrounds them by making use of devices and languages that are born in and influence our lives.

A well-aware student is less prone to the risks of remaining on the margins of the knowledge society, of the professional world and of suffering fake news.

Awareness also represents an essential skill when we talk about cybersecurity, for example. The problems that derive from the simple instrumental use of technology can affect personal data or predisposition to be bullied and require also more and more qualified personnel to solve the risks generated by cyber attacks.

Inclusion. It is one the great purposes of OP.

It represents the scope of making accessible a program to every student, regardless of their gender, ethnicity, physical or psychical disability.

Building an inclusive environment, especially in Computing Education is a demanding job, as it generates continuous challenges in order to assure the same opportunities to everyone in regard to their needs.

It demands to remove any possible obstacle to the process, and for this reason it guarantees the right to participation, the social engagement, the self-determination, both as individuals but also as part of a peculiar group.

The K-12 Computing Education design must adapt the settings, the didactic and methodologies to the special needs expressed by learners.

Inclusion implies an adequate planning, well trained resources and organizational flexibility.

Equity. It is the principle by which each student is helped and supported in receiving exactly what favors his or her opportunities to fully develop one's potential, given by the sum of educational and social elements.

Success or failure are two sides of the same coin and are not in competition: a few outcomes of the training process can lead to immediate improvement; others instead can act as a motivational boost to increase one's knowledge.

Equity represents an enormous challenge especially for educators, who must constantly question themselves, intercept the real needs of the students and adopt the appropriate strategies to remove the obstacles for the full affirmation of each one.

We speak about equity especially when the age range of a group is not coherent, or when the starting competences are at different levels from the beginning.

Accessibility. Universal Design for Learning and accessibility are two crucial features to ensure equal access not only for students with disabilities, but for all students.

Accessible design allows anyone to address the topic and scope of the activities, to get advantage from training opportunities, to interact with peers, to create quality contents based on one's own potential.

The importance of eliminating barriers to accessibility consists mainly in ensuring students with disabilities the adequate tools and applications to address technology challenges and also to undertake a path in computer science.

These barriers not only prevent access but also the full realization of people: it is precisely people with disabilities who benefit most from technology in their daily lives.

It is crucial that researchers consider these students not only as users of technologies, but also as co-designers, intercepting their educational needs and also to better improve applications for their advantage.

Collaboration. When students are engaged in group activities or otherwise involved in solving a challenge or a problem, they can be guided in achieving the common goal.

The learning process that is built through collaboration leverages the emotional aspects of individuals' participation to the group, but also on the personal skills that each of them can bring and share to enhance the result.

Through collaboration, the groups can experiment with job dynamics by assigning to each member operational roles and tasks for the development of the project work or the creation of a final artifact.

Furthermore, the team facilitates the development of relational skills, such as mentorship towards peers with less specific competences.

Empowerment. In the design of outreach, researchers often consider to include strategies and activities that can lead to the empowerment of learners, i.e. the ability of an individual to achieve self-awareness and control over one's own choices or actions.

This principle concerns many targets and audiences, for example the women not yet oriented towards STEM university careers, or underrepresented targets who come from rural or economically and socially disadvantaged areas.

Empowerment is the basis of both individual and group growth, it adds value to the school and later to the work path, it builds gender self-determination.

Creative Learning. Many K-12 education activities are based on, and practically use, creative learning features.

This expression defines learning methods and practices that exploit divergent thinking, the ability to re-elaborate knowledge subjects and to reflect on one's own personal and educational development path.

Creative learning welcomes errors and failures in the exploiting of a challenge as an indispensable step in the learning process.

The failure is not judged or sanctioned, but just accepted, analyzed and explained with the aim of restarting the experimentation after reflecting on the scope and consequences of the error itself.

It therefore helps in the process of reworking on learning and improving self-assessment.

As seen in Chapter 2, Seymour Papert was the first scientist to develop a real practice of creative thinking and learning, leading students to experience, even by trial and error, the maximum creativity [115, 116].

4.2.3 *Strategies*

In the realization of the programs there are also some strategies that can make them more effective and that recur in the scientific literature.

Pedagogy. Collaborative learning is the strategy that maximizes the student's ability to achieve a common goal together with his or her group.

Learning in and with the team through exploration, discovery and experimentation has, as a direct consequence, the enhancement of the individual contribution with respect to the group of peers.

It is an active approach to build one's knowledge but also to develop interpersonal relationship dynamics.

Team and Community building. If collaboration and participation are the basis of outreach, undoubtedly one of the most effective ways to promote them is the construction of an educational community around an activity, a methodology, or even a programming language or the use of a tool [61, 127].

There are various examples of educational communities that are both a driving force and a guide for novices, but also a support for those students who have already had some previous experiences.

The ability to build and conduct a community also lies in knowing how to exploit foundational principles such as collaboration, equity or empowerment.

Artifacts. The creation of a physical or virtual object at the end of a session represents a very important moment for sanctioning the participation of the entire group in the project.

There are many types of artifacts, from making a piece of software to building a robot or building a website.

The moment in which the object is presented and shared with a showcase constitutes a fundamental educational step for taking stock of the individual and work group's path.

The reflection that arises from the design of an artifact engages all participants in identifying strengths and weaknesses but also in developing a real communication campaign to share the results of the work [159].

Co-design and Customization. Co-design is a design complexity factor that can really broaden the prospects of an outreach.

Requesting feedback from participants in order to build and implement activities according to their actual training needs is an important element to ensure higher quality.

In the case of students with disabilities, however, the operators must know in advance the specific difficulties these students may encounter in the use of technologies and even in accessing and using the devices.

The activities must be planned taking into account these needs and any possible difficulty, and for this reason they are very rare because they are difficult to implement. Furthermore, the number of participants is statistically small and it is necessary to develop strategies for gathering the participants in the same location.

To recall one of the principles set out above, the inclusion of people with social disadvantages for example, we find initiatives that involve them using specific recruitment campaigns with a language appropriate to their context, but also with the support of local communities or agencies that already care and support them. They are often accompanied or persuaded by volunteers or social care staff.

Role-models. In order to enhance the participation of women, a common strategy is to promote the events by kicking them off with the speech of an IT woman who can share her story and represent a role-model for them talking about her career, motivation, her moments of failure and how she overcame them.

This type of speech is of enormous impact, because it demolishes many stereotypes that women suffer or are convinced of, and lead them to new perspectives on STEM and the role of women in IT professions.

Evaluation and lessons learned. Evaluating an outreach and measuring its impact presents different orders of issues, both in the design phase and in the course of data collection of the program's outputs.

It is somehow complicated to elaborate a standard or common evaluation process because of the strong heterogeneity of the initiatives.

PRINCIPLES	AIMS	STRATEGIES
Skills	Creativity	Pedagogy
	Persistence	TLE
Self	Gaps filling	Role-models
	Inclusion	Co-design
Interaction	Collaborative learning	Mentoring
	Sociality	Team building
Citizenship	Awareness	Community
	Empowerment	Self-Evaluation
Openness	Scale	Artifacts
	Reproducibility	Dissemination

Table 4.1: Design pillars of Outreach Programs

Dissemination. The outreach activities designed by researchers are often shared only with the scientific community and with a language that is not always accessible to a wider audience of educators.

Conversely, many activities carried out in extracurricular contexts by teachers and volunteers are not disseminated or publicly shared with the scientific community. The story of a project and its results can represent a starting point for the scale of a single initiative with other targets or audiences, in other geographical contexts or with other tools.

In Table 4.1 I resume the design pillars explained above.

4.3 THE DESIGN STAGES

In response to RQ2, I delineated the steps of the design process which take into account the above pillars, adapting them in the different educational contexts.

Further, I have classified the Systematic Literature Review in Chapter 3 according to the design attributes concerning educational and Computing Education research for outreach. I have also taken into consideration the socio-cultural features which distinguish these programs and which frequently mark their purpose.

The **8 design process stages** are:

- **Goals** - Identifying the educational needs and the target/audience, the pedagogy and approach, surveying the state-of-the-art;
- **Setting** - TLE adopted; human resources recruitment; location/environment arrangement;
- **Motivation** - Removing obstacles (emotional, social, infrastructural, logistics); role-models promotion;
- **Teaching/Learning** - Mentoring, peer-mentoring, collaboration, tinkering, self-regulation, self-evaluation;
- **Artifact** - Physical or virtual object, showcase and dissemination;
- **Feedback** - Pre-session or post-session harvesting; improvement of OP design; accessibility evaluation; co-design;
- **Evaluation** - Analysis of the impact on learning and interactions; use of the feedbacks;
- **Dissemination** - Sharing of results (publications, social media campaigns, learners' artifacts anthology); evaluation of scale and reproducibility.

In Figure 4.2 a graphic summary of the Outreach Programs design process stages.

Goals. The first design step involves identifying an educational purpose or goal.

This can take place according to different parameters: for example, one team of researchers can choose to submit a TLE to a group of participants to verify how they employ it or to collect their feedback on the approval or disapproval.

Sometimes the tool, for example a programming language or a platform, is literally tested by the participants, whose feedback can direct the follow-up analysis and further re-planning of the tool itself, or contribute to the co-design phase [3].

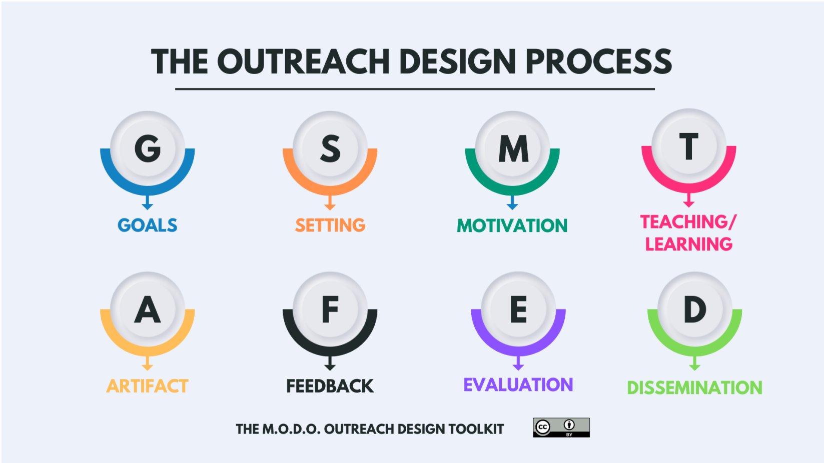


Figure 4.2: The Outreach design process stages.

Sometimes, however, researchers identify a specific audience to address the program. In fact, we have seen that there are programs aimed at women or underrepresented students who live in economic and social disadvantage, for example.

From the choice of the audience also derives the possible choice of some specific objectives to fill educational needs or some teaching strategies to encourage participation and motivation.

The design also takes into account the state of the art of educational research up to that moment and on those specific objectives. Some programs are the direct result of other experiments, from which inspiration is taken.

In the identification of the target or audience other elements also contribute, more linked to the organization area, which are specific to the experience of the researchers, or to the preparation of the human resources of the session, or to the possible presence of sponsors.

Setting. When designers opt for an in-presence program, a venue must be located.

Sometimes they already have an available space, as in the case of university researchers or companies workers, but often, especially for volunteer networks, it is necessary to ask companies, libraries, canteens for a space in which to operate.

The venue can be a decisive feature to ensure access: when it is close to participants, for example those coming from rural areas, it ensure success and avoids defections.

Coding clubs like CoderDojos usually mark the importance and impact of a capillary network in offering space for kids almost everywhere, even in remote areas or villages.

In choosing the setting, many other characteristics must also be checked, often closely linked to the purpose of the initiative.

For example a robotics program will require adequate spaces to allow the collaborative work in making robots.

The same also applies to the presence or absence of equipment and tools, as well as an adequate network infrastructure, if required by the session.

Motivation. The motivation of the participants is the fuel of an outreach program and also the real starting point for planning.

The primary feature of these programs, which differentiates them from the school, is their flexibility and relative reference to the curriculum.

This happens because the purpose of the OP is to promote participation as much as possible, encouraging the personal motivations or those of the reference group of the learners.

The identification of the audience and its motivations to participate must be carefully studied in advance.

The analysis of the literature or of previous experiments performed by the same team of researchers or similar to the one being designed can be of great support.

The motivations of some groups may be a reaction to cultural issues or stereotypes affecting for example a gender or an ethnicity. group.

There are programs designed solely to tap into these motivations and get to the heart of which is the drive to take part in a session.

This stage is often linked to the previous analysis of the perceptions that some students have of the subject: girls who have misconceptions about the role of women in STEM, boys from the suburbs with economic and social problems who consider themselves incapable of accessing to a certain type of knowledge and therefore of skills, for example.

Teaching/Learning. We have already mentioned the pedagogical difference that is created in a non-formal or informal learning environment.

Roles and hierarchies are mitigated and facilitation is the real teaching tool; the curriculum is above all a source of inspiration and teaching benefits from it but by adapting it to the context; the evaluation of learning is not translated into symbols, but analyzed by a community of participants, who exchange views.

The design of the learning environment and of all the stages in which to implement it is closely linked to the purpose for which the session is carried out but also to the availability of a structure in which to carry it out, of suitable tools or languages available, to the presence of trained personnel to follow the participants both in terms of new knowledge and the group dynamics that may occur.

Artifact - Planning an outreach also means predicting the results, which may be ephemeral or difficult to evaluate.

For this reason, an important element is represented by the analysis of the lessons learned, which usually base their foundations on the evaluation of tangible outputs, such as the creation of a software or a robot.

These artifacts translate the activities into objects, more or less material, to which part of the time of the session can be dedicated to create a showcase.

It is a moment in which the result of one's work is shown to others and it becomes important also because it requires getting involved and using communication skills to show and explain to others what has been achieved.

The artifact is the product of experience and therefore also of knowledge. It pushes those who learn to channel all the new acquired knowledge into an object and then to share its characteristics and the manufacturing process with all the others.

The showcase activities can be immediate or subsequent to the session, restricted to the group that participated or public.

Feedbacks. Feedbacks are a very important tool for collecting outreach results. They allow first of all to check whether the motivations of the participants have been satisfied or, if requested even before the start of the session, a way to evaluate the achievement of the program objectives.

Some feedbacks can be used to verify the achievement of certain skills, for example in programming activities or group activities, to verify if the components have worked with each other, attributing operational roles or collaborating effectively.

Based on the feedbacks, follow-up activities can also be designed including the participants in the process. Co-design thus becomes a further means of involving learners and increasing their motivation: planning increases engagement with the group, the activities, the final artifacts.

Evaluation. We have mentioned above which is the difference in the evaluation process of an outreach compared to the formal contexts, in which the assessment of students' knowledge and competences is standardized both in terms of methods and tools, and generally entrusted to teachers.

In the case of OP it could be more correct to speak of *lessons learned*, being that the evaluation of the entire experience and of how the entire group of participants behaved in rapport to the experience as a whole.

Dissemination. The dissemination of the activities is one of the cornerstones of the research activity: the experiments are described, analyzed and then shared with the scientific community.

The primary purpose is to reach other research groups to discuss and also obtain feedback on which to possibly reprogram new activities with a view to improvement.

Private companies that promote outreach held by their employees instead, tend to use their corporate communication channels, on which they publish the story and outcome of the activities also for the purpose of promoting their brand.

Groups of volunteers use web channels or social media to share the activities they have promoted and very rarely we can find detailed reports in the literature.

finally there are some platform, such as code.org [30], that also host learning environments or become themselves educational platforms, in which users can test themselves, alone or with the community of users.

Examples such as Scratch [138], whose platform and language are followed and managed by the community of participants, also coincides with the activities that promote it.

Boundaries of Outreach - We have seen how Outreach Programs express peculiar strengths, especially regarding the relational and social aspects of the participation.

However, it is possible to identify some limits of these activities, which emerge from an overall evaluation of the experiences. One of the main limitations of outreach, as already mentioned above, is the lack of adherence to or even the absence of a formal curriculum, as it happens instead in schools.

However, it is possible to identify some principles and objectives that guide the design of training experiences: even if they do not represent an educational framework nor are they highlighted as such by researchers, they can however be found in single programs.

Another element on which it is very difficult to establish a guideline or a rule is the evaluation toolkit. Many programs are oriented towards a mechanism of self-evaluation of the experience by the participants.

In regard to the assessment of the impact of OP, it is appropriate to distinguish two issues which have a significant effect: the duration of the programs and their reproducibility and scale.

Outreach are initiatives based on the involvement of audiences identified starting from specific training needs or from personal motivations, that are presumed to entice them to participate.

For this reason they are generally programmed in an episodic way, sometimes only once, for a minimum duration, from a few hours to half a day or a single day; other times they are camps, therefore with a slightly longer duration over the space of one or two weeks.

Often, they are not reproduced in subsequent periods, nor is possible to scale, adapting them to another target or audience. These boundaries constrain researchers, except in the case of initiatives with a large number of participants on a regional or national scale, from carrying out longitudinal evaluations, i.e. over several editions, even in subsequent years, and to obtain significant samples for impact evaluation.

Remarks. In this chapter, I have outlined the main strengths of the general criteria and pillars of the Outreach Program design.

In response to RQ1, I can affirm that a classification can then be defined. It is a general approach to the problem and, while the criteria are broad categories which qualitatively, and according to their purpose, define the possible phases of a project, the latter instead represent the specific instances in which the project is declined and delineate in a more granular way the different educational perspectives according to learning contexts.

The taxonomy is therefore a first step in defining the main categories and features of the programs differentiating them with respect to the principles, objectives and strategies in which they are then implemented. Therefore also RQ2 finds a positive answer. However, we have seen that it is also possible to highlight boundaries, or even actual limits, to educational research and to the evaluation tools available to researchers.

In Chapter 5 we will introduce a proposal of a design toolkit based on this taxonomy.

5 THE OUTREACH TOOLKIT

As seen in previous chapters, Outreach Programs represent an alternative and unconventional - or at least less conventional- environment for the learning process in Computing Education.

They operate as a complement to the school CS curriculum (see Chapters 3 and 4), and their nature is usually temporary, occasional, episodic, less structured in respect of ordinary classroom activities, (human) resources, organization and methodology.

In Outreach performance we rarely find the conditions for long-term programs which ensure a formative continuity, allow an in-depth study of design-related issues, or on which to carry out a longitudinal impact evaluation.

In spite of their tangible heterogeneity, however, it is possible to determine several average design principles, as seen in Chapter 4, common to most programs.

Given that in literature there are examples which lead us to detect these bases, nevertheless not all programs can adhere to these guidelines and sometimes a few crucial matters arise, requiring researchers to identify new design solutions. I am aware that the research can be extended and deepened, but I am also conscious that to obtain a further implementation of the outreach design principles it is necessary to trace some common starting points.

The definition of the major design matters drawing from the literature generated a taxonomy of designers' plans during the elaboration of a program. In this chapter, in order to design a practical toolkit for researchers, educators and practitioners, I will focus on some solutions to common problems which have proved to be more effective.

The most ordinary concerns can usually be faced on-site with temporary solutions and later embedded in the re-design stage, but it is also crucial to foresee them and, during the planning phase, to adopt some practical inventions which can support the effectiveness of the initiatives.

The solutions presented in the Outreach Toolkit proposal are not definitive or comprehensive of every aspect of the design, but instead I intend to move a step toward the scientific community to enhance a reciprocal exchange and collaboration.

Because of the breadth of the purpose, I have confidence and will appreciate if my analysis on the design of Outreach Programs in K-12 Computing Education could generate further questions and answers.

5.1 RESEARCH QUESTIONS AND METHOD

The most common matters in designing an Outreach Program are usually previewed in advance by researchers; anyway, sometimes they realize what affected the efficacy and results of the session only after a feedback survey, or if they directly face the problems on-site with temporary answers.

In fact, the realization of previous programs suggests researchers and educators several instructional or even practical solutions to opt for during the planning phase, which can support the effectiveness of the initiatives.

Accordingly to the above considerations, my contribution to determine the importance of a **design toolkit** is based on two research questions (RQs):

- **RQ1** - What are the major design concerns in OP?
- **RQ2** - What are the featured criteria on which drawing guidelines of a toolkit for the OP designers?

In response to RQ1, the steps followed in defining the major design concerns generated a classification of designers' precautions during the elaboration of a program.

The design concern taxonomy suggested in the present dissertation is the result of the Systematic Literature Review I conducted on Outreach Programs [5], the K-12 CEd Taxonomy of TLE [6], other significant studies on out-of-school K-12 Computing Education [**outreach**, 12, 16, 39] and on studies about the use of theory and design in Computing Education [94–96, 109, 148–150, 153].

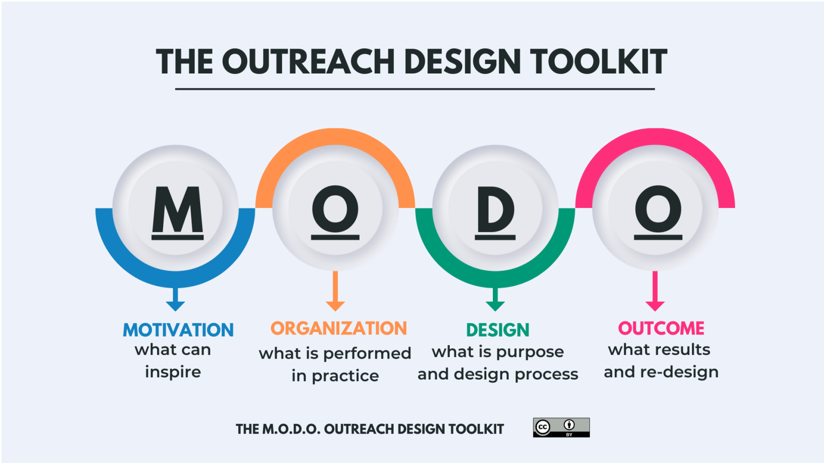


Figure 5.1: The Outreach in Computing Education Design Toolkit.

5.2 THE OUTREACH DESIGN TOOLKIT

The design of K-12 Computing Education Outreach Programs takes different shapes depending on the context in which it occurs.

Identifying the educational matters and their possible solutions, planning the activities, defining the methods for evaluating the outcomes [12], they all represent facets of a cross-disciplinary process and methodology that support the invention, conception and implementation of the initiatives.

General criteria. In response to my research questions, after surveying the major concerns for every pillar in Chapter 4, I outlined an OP design taxonomy along four criteria, which incorporate the different areas of interest to consider while designing a program:

- **Motivation** - *what can inspire.* It includes the incentives envisioned by designers that can influence or enhance the participation of specific categories of learners, boosting their access to computing literacy. It is also the set of personal expectations, desires, perceptions and stereotypes that students have with respect to STEM disciplines and Computer Science in particular;

- **Organization** - *what is performed in practice.* It includes the operational instances to settle the educational and social environment in view of the implementation of a program;

- **Design** - *what is the educational purpose and the design process.* It concerns the planning, the elaboration phase and the solutions adopted for the initiative, with regard both to the educational features and the computational ones.

- **Outcome** - *what is the result and can support re-design.* It concerns the forms and quality of the results, they are tangible or not, of the programs and supports the re-design stage.

The four criteria are listed according to the order and frequency in which they appear in the design reports and are displayed in Figure 5.1: their M.O.D.O. initials remind also the Italian term which stands for “mode" or “method".

Generally, in literature, the researchers express first of all their choices and intentions on how to address the motivation of the participants and how they mean to promote it during the sessions. Later, they present how they designed the program and how it has been implemented.

Furthermore, some programs are mainly guided and determined by the rise of practical issues that must be resolved before moving on to the actual design phase, because otherwise they could hinder the implementation itself.

Consequently, the third criterion is closely related to the first because it represents the translation of the personal and social needs of the participants into concrete goals, approaches and activities. The outcomes are results, but somehow also stepping stones to the further development and re-design of a program.

Motivation. The main area where designers seem to have directed their efforts is the one of the ambitions, right or wrong perceptions, prejudices and stereotypes that some groups of students have towards Computer Science and STEM disciplines as a whole, before adhering to an OP.

In fact, the first group of concerns is actually about the motivation of participants and the strategies to foster it. When researchers in K-12 Computing Education attempt designing a non-formal or informal program, they reflect on which spark could better seduce and attract an audience.

As seen in the survey in Chapter 3, by *audience* I mean a cohort of people with a common connotation or attribute: the gender, the social or economic affinity, the physical or mental impairment, and so on.

Motivating people differs from simply training them; it is more properly a strategy to take into consideration human expectations or even misconceptions rather than designing a general framework or a course curriculum suitable for everyone.

It is an educational investment in addressing the resolution of cultural, social, economic and behavioral factors that hamper self-consciousness and fair ambitions.

The gender gap, for instance, is taken into great account by K-12 Computing educators and researchers from a few decades. Programs addressed to women concentrate on enhancing the participation of female learners in Computer Science and to promote their choice of university careers in STEM disciplines.

Increasing girls' interest is a matter that involves many cultural factors usually affecting their participation, originating from society stereotypes or family clichés about women's role in technology, the disapproval on female ambition and the guilt on self-determination [50, 59, 70, 75, 85, 128].

For the same reasons, in the gender gap issues must be included those referring to whole range of the various gender instances, not just those of women [34].

Engaging underrepresented genders in K-12 Computing Education is arduous because stereotypes are often hard to overcome and interventions are difficult to organize and perform. Sometimes it is girls' previous under-confidence that affects participation [128].

Some programs become efficient only after a long-term action, that gain influence on socio-cultural bias [28, 47, 85, 101, 157].

A useful and recognized trigger is the introduction of a *role-model* during the session, especially at the beginning.

A role-model in gender gap programs, for example, can be a computer scientist who demonstrated persistence and pursued success in CS against gender barriers and clichés, breaking the rules and finally sharing her path with other young women.

She explains the obstacles she had to face and the achievements she definitely conquered. Girls can identify themselves in the story told, and scale it to their own journey.

Role-models are an effective instrument especially to overcome women's fears or wrong perceptions about their real possibilities due to the socio-familiar heritage that prevent them *a priori* from Computing Education and IT knowledge.

The introduction of a role-model depends on the availability of a scientist with a specific experience on the topic of the proposed program. Stereotypes can be broken with these amazing living examples which speak of possibilities and results, avoiding the risk of the "imposter syndrome" [65].

Another approach that is effective with women is the collaborative learning and the support to teamwork when developing a software or an application, building a robot or a website. These activities boost and consolidate the interactions between the participants, supporting the self-attribution of operational role in the teams and the decision-making capacity among the various figures in the groups [155].

In fact, it is no coincidence that one of the most adopted outreach formulas to involve girls is the competitive one, such as a hackathon or a contest. During these sessions, the girls measure themselves in teams, against short-term but highly satisfying goals that increase their self-efficacy and awareness of their own role and attitude in computing.

The initiatives that generally promote access to programming, i.e. sessions for novices to introduce them to the basics of Computer Science, put great regard in widening participation to all targets and audiences.

As seen above, some examples involve specifically women, but many other initiatives, likewise, address underrepresented segments of the population, especially young learners coming from disadvantaged social or ethnic contexts [24, 48].

Under this concern we comprised also the inclusion matters, which are supported by rare programs to broaden participation in K-12 Computing Education to kids with disabilities or special learning needs [90].

The purpose is to guarantee equal access to computing: these groups of kids need a well designed proposal due to the mildness

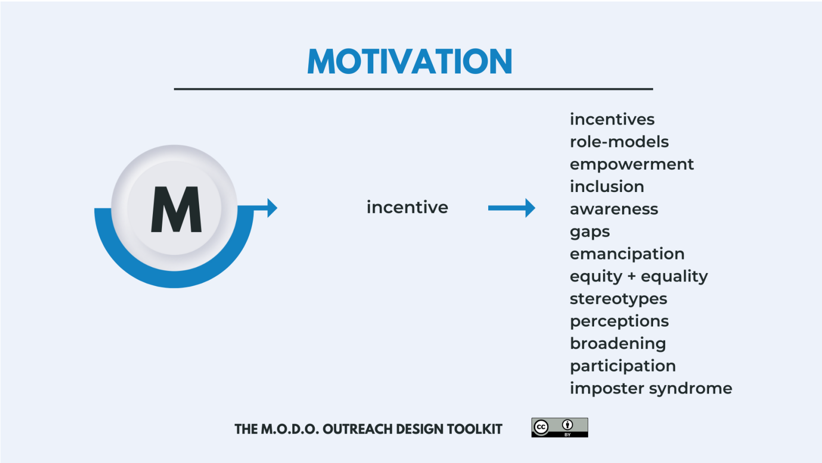


Figure 5.2: The Outreach Toolkit: Motivation criterion.

of the different approaches and the various levels/difficulties they can encounter in traditional settings or environments.

Equity is well promoted especially in learning environments where a community is active and members support each other.

When novices enter the community, they are supported by mentoring strategies and so they can share their knowledge and progress in computing while at the same time they learn by social interactions [7].

The best and renowned experiences connect the use of a language to an environment and a community of learners [37, 61, 98, 127].

In Figure 5.2 a summary of the features and keywords of the Motivation criterion.

Organization. In this asset I included the solutions to practical issues and concerns that guarantee an adequate environment to participants and the feasibility of the program.

Outreach can be performed in presence, in the form of camps, after school sessions, coding clubs.

Otherwise they can happen online, especially when distance represents the crucial element that affects the organization and implementation of the initiative. therefore, remote participation becomes an enabling factor for underrepresented categories, as rural area students [32, 42], who often cannot afford long trips to reach a venue, or even in case of extraordinary circumstances, as seen during the COVID-19 pandemic.

The choice of the **modality** (in-presence, online) allows learners to participate or not, and can really change the design of the program. It should be the first step to define, as it affects the practice and otherwise it would be very difficult to reach those who cannot join for the above reasons.

It is intended that, in absence of a good Internet infrastructure, it is impossible to consider the remote modality to ensure participation.

When designers choice for the in presence program, a **venue** must be located. Sometimes promoters already have a suitable place at their disposal, offered by an institution or a sponsor, as in the case of universities or companies but often, especially for volunteer networks, it is necessary to ask companies, libraries, canteens for a space in which to operate [108].

A venue which results close to participants and really accessible will improve the participation to the initiatives and to avoid defections. Coding clubs like CoderDojos usually mark the importance and impact of a capillary network in offering space for kids almost everywhere, even in remote areas or villages [122, 141].

Outreach need **tools**: they can be physical or virtual instruments, but also languages and environments that designers can select for the programs [6, 102, 103].

Some initiatives are planned starting from a single language or programming environment, around which a community arises [132] and later the program is promoted to learners as a meetup.

Other programs, like robotics sessions, need well trained **personnel**. Sometimes the tools can represent a boundary to the design and implementation of a program, especially if they entail high costs too hard to be faced by the organizers.

This happens frequently in the case of robotics sessions, given the high value of the devices. When using a programming language instead, the organizers should consider a previous download or setting on each device before the session starts [136].

If the environment is totally online but participant need to register a personal account on the platform, it could be an advantage to suggest kids to do it before joining. In both cases, a human resources training is essential to guarantee a proper mentorship during the programs [35].

Outreach initiatives differ according to the overall **duration** of the projects. Some programs are one-time sessions that last single slots of few hours, like the coding clubs; some more can be planned in several days, like the summer camps, where there is a minimal progression in the knowledge acquisition with activities bridging to subsequent steps; some others can last a few days, even in non-stop sessions, like in the hackathons.

The duration of a program affects its quality and impact on the community of learners. Short time sessions are more suitable for an essential literacy or to promote a context of experimental creativity; on the other hand, longer slots can help in digging deeper into a topic or a language.

Then, the **scale** or even the **reproducibility** of a program [79] depends also by its duration. It determines the choice of the same organizational solution by designers, when they evaluate other factors, like a disposable venue or available operators.

Costs are another crucial element: while a company can easily decide to invest funds in social activities with a certain regularity, volunteers without funding instead are often forced to resize their idea to the less expensive one, or to undertake a fundraising campaign [141].

Teachers, mentors and operators are an essential component of the outreach programs. They act as **facilitators**, leaving kids the time to discover and tinker, but when they are well trained, they can also add a peculiar value to the initiatives.

This is crucial when the program request a certain proficiency, for instance in robotics camps [91], where specific skills are necessary, or when using a programming language that need proper knowledge. Sometimes CS researchers recruit undergraduate students to mentor with middle or high school kids.

A good practice is to involve the senior participants in training juniors to provide an eventual turn-over. **Recruiting** needs empathy, a good relational competence and abilities in communication via periodic calls.

It is also important to evaluate the possible costs of the resources. Companies can count on their own workers, but vol-

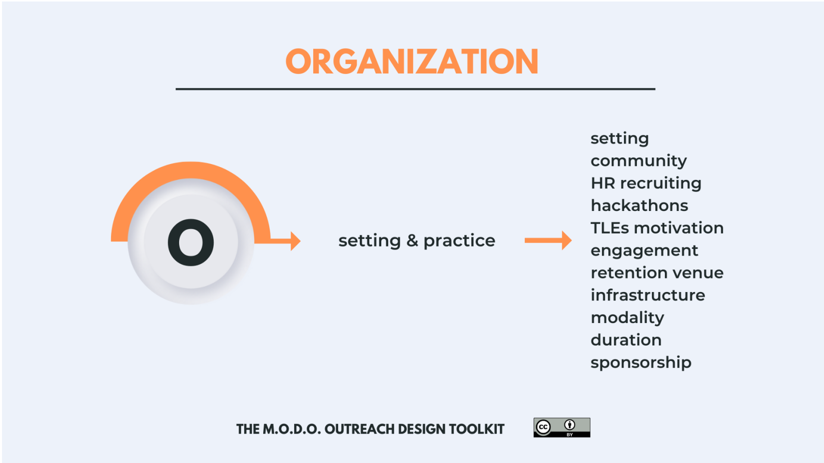


Figure 5.3: The Outreach Toolkit: Organization criterion.

unteers are usually not paid and this eventuality can affect the continuity of their participation in the long-term.

Communication is a crucial concern when designing an informal learning program. It has two directions, internal and external.

Internal communication pertains to all the processes of organizational information to participants, operators and mentors, sponsors and venues owners.

The other layer refers to the **dissemination** of the initiatives, from a website to a newsletter to social media or messaging apps.

Communicating with different age groups or audiences can be difficult and it needs instruments without barriers in terms of accessibility [110].

The tools for engaging a good communication with participants can be adopted also to call, advise or recruit operators. If well organized, the channels can be used also for purposes of institutional/formal communication in case of national/international code weeks [108] or of dissemination of the outputs.

It is crucial to determine if a program is sustainable on the economic perspective, because its long-term duration, or even scale, is influenced by this factor.

As already said above, the problem of the costs can really affect programs and determine if their implementation can happen or not. While many initiatives can rely on a sponsorship from a company or a public institution, more often several informal programs originate from spontaneous groups of volunteers or non-profit associations that don't have any funding.

Some practitioners, for example volunteers in coding clubs, can ask parents or attendants to contribute with a little fee to sustain the major costs, like providing tools or rent a proper location for a limited time.

In Figure 5.3 a summary of the features and keywords of the Organization criterion.

Design. The design stage of an Informal Learning program benefits from the two previous assets for the selection of the organizational items and the motivation issues researchers intend to solve.

Designing an effective environment has many concerns, constraints, and possible problems. It is probably the most theoretical asset among the four, therefore it is also the most challenging [43].

Designers should evaluate in theory, in advance, what they want to realize later in practice, especially before selecting a tool, a language, or a technology for their session.

They should evaluate universal principles, like equity and inclusion, but also the knowledge level they want learners to achieve [137].

Good design can configure a sound success to the initiative: scaling a program means that it turned out flexible and adaptable to other targets or audiences.

The evaluation of the impact support the possible rearrangement of the design and must be made on most of the concerns we presented until now.

Co-design is another resource, alone or in synergy with master design: it can involve participants at different stages of the planning [1, 126], or even after program implementation. It is a crucial method to ensure the best user experience and consequently also to increase the impact and reach of the program.

Besides the theoretical principles, designers determine the pedagogical approach that inspire them and they want to adopt in the programs.

As said above, these two concerns, i.e. design and pedagogy, usually overlap or are at least complementary. There is a vast literature on the pedagogical approaches in Computing Education, as seen above in Chapter 2, but Outreach Programs are essentially learner-centered.

As the role of adults is reshaped, a form of tutoring is a worthwhile feature: in a less formal context, educators are mainly facilitators of the learning process. Learners can better participate when some coaching strategies are implemented in the session [60].

Mentoring can be performed by adults, or even by senior participants in the form of a peer-mentoring. These practices prevent drop-out effects for kids who struggle, facilitate interaction and networking abilities of participants, and balance gaps between the diverse backgrounds, harmonizing issues and solutions and boosting motivation.

The main agents of Outreach Programs are definitely learners. The participants can be selected by target, meaning a consistent group by age or school level, or by audience, with a social, economic, gender or impairment attribute.

A program is performed in a setting where kids can interact and collaborate, often working in groups [71]. Therefore, the efficacy of a session is closely linked to the social interaction and cooperation. Learners are immersed in a meaningful environment, supported by peers and mentors, and they can achieve goals in line with their aspirations, with an approach of connected learning [64].

The sub-groups can work on different tasks or projects, and this modality determines the collaboration mode of participants.

Sometimes a program can be designed for a competition and for the goal of a prize achievement, for example during hackathons: the groups cooperate to the success of a project, creating a physical artifact or developing a software, but at the same time they contest against other groups in view of a final award.

Evaluating an Outreach Program is often difficult due to its unconventional nature, the limited duration of the sessions and, in many cases, the non-reproducibility.

Some designers plan several repetitions of the session with the same target or audience; some others schedule a number of

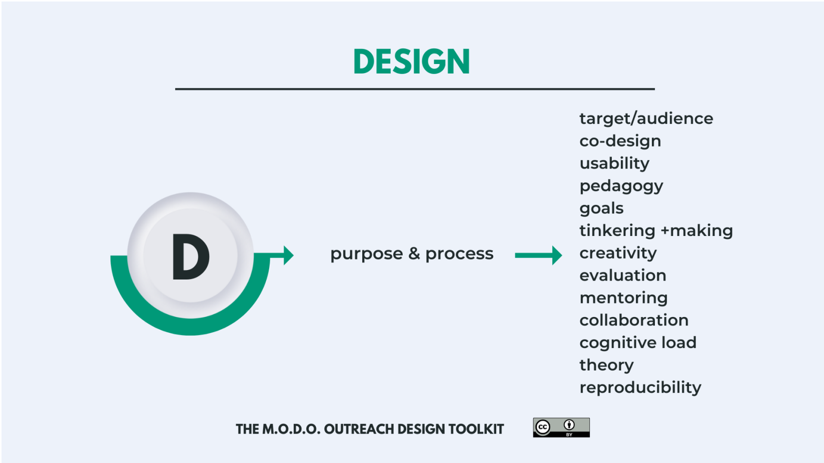


Figure 5.4: The Outreach Toolkit: Design criterion.

meetups with a progressive development of the topic steps and goals.

A few initiatives have been surveyed and analyzed under the assessment constraints, to determine if it is possible to define a general theory, helpful for the design or re-design of the programs, and establish how to validate the participation [86].

The shape of the outputs can be predicted at the design stage, because they often highly engage students in a crucial effort and desire to showing the community the result of their commitment.

Outputs benefit of the pedagogy and design of a program, but they also influence the assessment, successive updated versions and scale of new initiatives [82, 86].

The participants' feedback, the final display of the projects and the tangible artifacts represent a crucial contribution to the outcomes. They can be collected, or even presented during a final showcase, where kids share their works with the whole community, peers and mentors [140].

Evaluating an outreach and measuring its impact present a different order of issues, both in the design phase and in the course of data collection of the program's outputs.

It is somehow complicated to elaborate a standard or common evaluation process because of the strong heterogeneity of the initiatives.

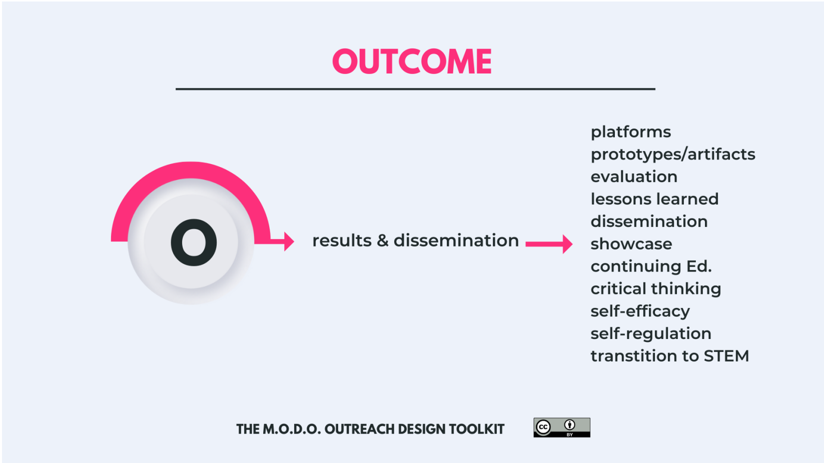


Figure 5.5: The Outreach Toolkit: Outcome criterion.

The problem still difficult to face and less solved in the field of Informal Learning is the evaluation of the learning outcomes. In fact, it is hard to assess on scientific grounds the effectiveness and impact that this approach can generate, because it is often linked to methods of implementation and fruition that are very complicated to arrange. Regarding the evaluation and possible certification or validation of the outcomes, by the way, there are some interesting institutional frameworks [143] and scientific contributions that can inspire designers [39, 86, 137].

In Figure 5.4 a summary of the features and keywords of the Design criterion.

Outcome. Designing a program requires also to imagine or foreseen the outcomes, especially in terms of artifacts and impact. On the one hand, these two elements represent the goal of the activities, on the other they can be considered as the real purpose.

Some sessions are aimed, by their very definition and the approach they adopt, at the creation of an artifact. In hackathons, for example, researchers foresee in advance that teams will compete to win a final prize by creating an object, physical or virtual, which will be evaluated by a jury selected for their specific experience on the themes of the competition.

The artifact is therefore practically the starting point on which the activity is designed, the pedagogical approach is evaluated, the audience is selected and so on [12, 140].

CRITERION	FUNCTION	AIMS
Motivation	Incentive	Enhance participation Boost literacy Personal expectations
Organization	Practice, setting	Environments Human resources
Design	Purpose, process	Planning Evaluation
Outcome	Dissemination	Artifacts Lessons learned Dissemination

Table 5.1: General criteria of Outreach Programs design

The outcome is also the starting point of the re-design. It provides feedback from the participants, elements that can also be used in co-design sessions with the students. They can give an overview of their appreciation and acquisition of knowledge and skills, both on the chosen theme or tool, and in terms of built relations or collaborative work. The results can also be used in conjunction with designer tracking data if it has been collected.

In Figure 5.5 a summary of the features and keywords of the Outcome criterion.

A summary of the four criteria, their functions and some of the major aims are listed in Table 5.1.

Design concerns. The design of the Outreach Programs is a complex process that involves researchers in finding the best proposals and solutions, both from an educational and organizational point of view. Planning and subsequent practical implementation sometimes collide with unforeseen matters, or problems that occur at the moment.

In the design phase these concerns can be expected or even anticipated with strategies, developed to deal with them in the implementation stage; when instead they occur during a session, unless they hinder its implementation, they can be very useful for making a detailed analysis of the impact that a program had on the participants or in any case on the purpose of the initiative.

In order to guide the analysis, I adopted a personal categorization of these concerns by integrating the taxonomies developed above in Chapter 4, such as the motivation, organization, design and outcome criteria, but also the three design process pillars of Principles, Aims and Strategies.

The result is a set of **five designers' concern clusters**, as follows:

- Literacy;
- Advancement;
- Gaps;
- Periphery;
- TPD;
- Impairment.

Literacy. There are many initiatives to introduce novices to Computing Education and in particular we find many programs that focus on Computational Thinking and programming in its different forms (unplugged coding, digital storytelling, robotics and physical computing, game development...) as the first approach to raise a computing literacy [4, 87, 92, 151].

The choice of a programming language or environment looks crucial to determine the success of the program, if well valued and selected for the specific target or audience. For example, when speaking of targets equivalent to school grades, there is a general tendency to prefer a block-based language for primary school children [127].

On the other hand, for older students this choice tends to be avoided because it is often not accepted by the participants since it seems more suitable for a younger age group [61].

In addition, middle and high school students already have certain personal expectations or desires, for example towards robotics, microcontrollers and game or web development, and researchers adopt these suggestions in course design.

There are also activities designed for the professional development of those teachers who do not have an IT background but who regularly use devices and tools in teaching, as well as exploiting programming for interdisciplinary teaching.

Advancement. Outreach Programs can also represent interesting opportunities for the growth and advancement of computer skills, both for students and teachers.

Regarding the former, we generally speak of those programs that are aimed at the so-called prospective students, i.e. at young people who have already decided to continue the pipeline by choosing a university career in the STEM field, or at those who didn't yet, but who potentially could still mature the decision.

In these cases there are already several prerequisites that determine the planning of the activities for the growth and enhancement of knowledge and skills.

Teachers can also be involved in advancement and growth programmes, particularly if they are already IT teachers who wish to update their knowledge and skills for an immediate employ in their teaching practice at school.

Gaps. As we have seen above, there are various kinds of gaps, mainly due to the students' belonging to a gender, ethnicity or socio-economic environment.

Each of them spark challenges and problems for researchers, who study how to solve them. For example, when involving women there are reasons of family and socio-cultural nature that affect their participation and above all the motivation towards STEM disciplines, weighed down by the stereotype that considers them for men only.

For underrepresented groups, such as students experiencing social disadvantage, often the main problem is how to reach and involve them, because they dodge outreach calls or the specific educational proposals for them.

Some initiatives focus exclusively on the social interaction that can be created during the sessions, aiming to fill the relational gaps that these students experience in real life.

Periphery. Living in peripheral or disadvantaged areas is one of the problems that designers ask themselves when choosing and audience, because they can have consequences on the frequency of programmes.

Sometimes the geographical distance from the venues where the outreach takes place affects or hinders participation.

The "periphery" concept that I adopt here had a strong development, even metaphorical, during the COVID-19 pandemic when, in order to reach students, it was necessary to intervene by changing delivery methods and switching to online sessions.

Rurality is a more specific concern than the previous one and it regards those students who, in addition to living in marginal areas afar from the venue where the outreach takes place, also belong to rural social contexts, in which it is very difficult to apply the knowledge learned or further develop their skills due to the lack of IT devices or adequate technological infrastructure or internet connection.

Therefore, rurality is an overlapping of social and geographical issues with those of the digital divide, both for students and their families.

TPD. In the perspective of maximizing student literacy as well as their advancement and growth in STEM, some K-12 Computing Education programs are specifically addressed to teachers.

Indeed, those who teach in school but also outside school, are the ones who can support and facilitate the learning process of computing. However, one of the main concerns coming to the observation of designers is represented by the early preparation of teachers.

In fact, not all of them already master the subject or are STEM graduates, sometimes they are experts in other disciplines who need to train starting from the basics, so they are novices with a limited experience or they rather need computing literacy from scratch

On the other hand, STEM teachers who already teach these subjects in their classes need a solid update on technological innovations and a professional development on advancements in teaching, in order to make it effective.

A common concern of designers, when they work on teacher training, is the teachers' worry to get involved with new things and to confront themselves with colleagues.

Adults show more fears and less lightheartedness in their activities, they feel judged even if the context puts them at ease, they often struggle to understand how much their contribution is worth or significant or effective in a group work.

Impairment. The number of programs specifically aimed at students with disabilities or from physical and psychical minorities is extremely narrow and it might seem counter-intuitive to mention them.

However, as we shall see, these programs pose some interesting and relevant concerns to designers. First of all, this group of programs marks an interest in these learners' categories and generates accessibility and usability issues for sessions and activities.

The main factor emerging from the literature is the small number of participants [5]: the reason is due to the challenge to gather together, in the same session with a specific purpose, participants with the same type of disability or impairment and who concurrently also have the same skills and initial knowledge, or who come from homogeneous social backgrounds.

Another problem is represented by the real capability of learners to use some technological tools in relation to their disability. It therefore happens that for these programs researchers reflect mainly on usability issues and ask themselves how to choose or adapt the tools to both the practical and educational needs of these students.

Finally, a relevant aspect that emerges from reading the experiences already achieved is that the Outreach Programs for these categories of students offer a proportionally lower level of knowledge and skills than that of their peers, as a result of the fact that they access STEM later, with a certain delay mainly due to the lack of courses specifically designed for them.

For this reason they represent, even in their exiguity, a specific group of experiences and activities to be taken into consideration for the value they add to the inclusion matter.

In response to my research questions, I surveyed and matched the major design criteria and concerns and attributed them to every above pillar. They are briefly summarised in Figure 5.1.

An issue while designing and managing OP and not included in the toolkit is the evaluation of the learning outcomes. In fact, it is hard to assess on scientific grounds their effectiveness and impact due to their sporadic nature (duration constraints) and because

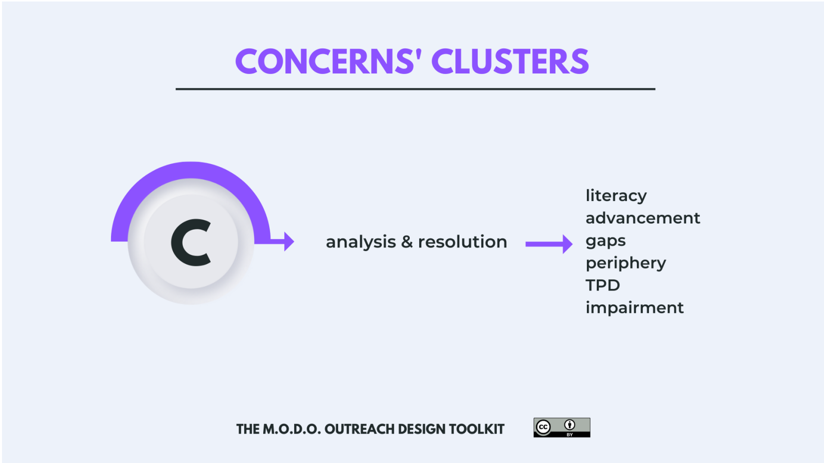


Figure 5.6: The Outreach Toolkit: Concern Clusters.

the methods of implementation and fruition are extremely variable. researchers themselves prefer to speak of "lessons learned", meaning that thy can just evaluate the programs one step at a time and in respect of the experience already performed. Regarding the evaluation and possible certification or validation of the outcomes, by the way, there are some interesting institutional frameworks [143] and scientific contributions that can inspire designers [39, 86].

In Figure 5.6 a summary of the Outreach Concern Clusters.

Remarks. In this chapter I have first of all described the design phases of the outreach, specifying for each of them the development and the precautions of the researchers.

Subsequently, answering to RQ1, I outlined the main concerns which occur during the design or follow-up phases to program designers.

The description is based on a taxonomy of the areas of interest in which the concerns fall, which also coincide with some areas of design scope.

As far as the RQ2 is grounded, from the analysis of the concerns it was actually possible also to further devise and trace the framework relating to the possible design guidelines, for the use of the researchers.

In the toolkit I preferred to indicate a certain number of solutions, both for the elaboration of the activities and for their implementation, with suggestions that could be useful in the practice.

The toolkit aims to support researchers in the OP design but can be further implemented, as seen above. It is a start to reflect on the concerns regarding the planning and implementation of the programs.

Part V

CONCLUSIONS

Collateral learning in the way of formation of enduring attitudes, of likes and dislikes, may be and often is much more important than the spelling lesson or lesson in geography or history that is learned. For these attitudes are fundamentally what count in the future. The most important attitude that can be formed is that of desire to go on learning.

John Dewey, *Experience and Education*

6 CONCLUSIONS

The role of the teacher is to create the condition for invention rather than provide ready-made knowledge

— Seymour Papert, Mindstorms

Digital technologies shape our society in a disruptive way and have accelerated its development and evolution.

This incremental progress is reflected in every area of our lives and especially in Education, because it requires new and complex knowledge and specific skills in IT.

The impact of this transformation also occurs in our relationship with Computer Science and technology, no longer just as passive and instrumental users, but increasingly active creators of contents, tools, languages, networks, communities.

The educational pipeline is adapting itself to this profound evolution and the school for its part is affirming the importance of ensuring digital literacy and training to students.

In several countries, the formal curriculum is expanding with Computer Science fundamentals and IT knowledge to enhance everyone's social, economic and civic participation in the society.

The crucial value of this type of training as a factor of change and personal empowerment affects the inclusion of each citizen and the global progress in every area of our lives.

Technology facilitates the educational process and, over the long term, with its own tools and languages, guarantees continuing education both in study and work.

Computing knowledge also determines social driving consequences on citizens' self-determination, especially for underrepresented categories or those who experience digital gaps and social exclusion due to ethnicity, gender or physical impairments.

Nowadays, the lack of awareness or expertise in CS represents not only a cause of educational deprivation, but also a reconditioning feature from family and society.

Computing Education is the relatively young and rapidly developing field which addresses the demand of technology education and the construction of computing knowledge and competences. It responds with an inter- and cross-disciplinary attitude, with researchers coming both from the perimeter of Computer Science and Pedagogy. The inevitable and necessary contamination between the two groups generates a few increasingly crucial reflections, both at a theoretical and epistemological level and on the practical implementation of the experiments.

Outreach Programs in K-12 Computing Education are a consolidated reality that is constantly being carried out besides or in parallel to the formal curriculum, and having a formative and social impact on participants. They are learning environments which integrate elements of the computing school curriculum with the motivation and social needs of the participants and that base their action on the incentive to participate, collaborate, experiment with languages and tools, filling gaps and digital divide.

Even if it has an unconventional and sometimes incidental approach to learning, computing in outreach expresses a variety of environments, scopes, and methods similar or complementary to the formal ones and it needs good design to implement the programs. In planning the initiatives, there are a few constraints and matters that researchers usually must face and solve.

6.1 DISSERTATION CONTRIBUTION

In my thesis I have outlined the process and what are the major concerns in the design of an outreach session in K-12 Computing Education, with the intent to build a design toolkit for researchers and educators.

In relation to the four areas of intervention, I illustrated which are functional to the motivation of learners, and how to engage and motivate kids; I also defined several organizational issues and solutions; and finally reflected on some major design prompts behind the planning.

The intent was to confirm the value of these initiatives and show that, even in a non-formal or unstructured environment, there are several common design strategies that it is possible to adopt and implement, and that there are also possible solutions to solve the most frequent issues.

The taxonomy of these features is built on previous literature and on the concerns I consider more sharp in the design of the programs. Obviously, I am conscious that our discourse can be implemented and that there is still room to create analogue classifications in regard to the pedagogical objectives, the social issues of the target and the organizational matters, for instance.

Researchers often have to deal with some design concerns and face some specific challenges during the implementation of the programs. The design process, which is described here in its stages, in this dissertation is also enriched by a proposal of an operating toolkit which has not yet been found in the literature.

The suggested solutions meet most of the design concerns, but I have also highlighted some instructional constraints that are not resolved at the near moment, such as the impact assessment of the initiatives.

The toolkit may also have an utility for the practitioners not directly enrolled in research or education, such as the volunteers or companies' workers who carry out informal outreach sessions. In many countries where there is no CS curriculum, their camps and clubs actually offer the only meaningful access to Computing Education for many students.

The Outreach design toolkit aims mainly to support researchers, educators and practitioners in planning the Outreach Programs. For example, many courses take place in contexts such as summer camps or coding clubs, where volunteers happen to be engaged in solving more strictly pedagogical problems.

The toolkit can help, both during the design and the follow-up stage, for the resolution of the most common and frequent problems, providing guidelines in the single project areas highlighted with the above criteria.

6.1.1 *Future directions*

A possible development of the Computing Education Design Toolkit M.O.D.O. could concern its application with educators for TPD, in which relations and self-regulation weigh on the results. In this case, the adults' difficulty in getting involved and sharing one's learning path without hesitation, essentially reversing one's role from teacher to learner, has a considerable effect.

Another important future development that might be taken into account and that could be worth investigating is the assessment of the outreach impact, above all by identifying a qualitatively effective system to implement it.

Finally, a useful contribution that can represent a future direction to this research work is a more comprehensive definition of a taxonomy of TLE. In our research lab I am already working with other researchers on the further implementation of our collaborative Computing Education OP Taxonomy platform on GitHub, open to the contribution of worldwide scholars, as seen above in Chapter 3.

In the end, it would be interesting to foresee a validation of the educational framework and toolkit with some experiments aimed at solving the highlighted concerns.

For example, one of the possible experimental activities could be carried out with a group of researchers: it could be proposed to use the toolkit to create an experience with a specific audience and then the related evaluation.

The verification of the practical usefulness of the guidelines could also be validated in an informal context with a group of practitioners, such as for example the volunteers of a coding club, in the organization of a session. In this case the data collection should be done by the researchers, but the context and the audience could provide very interesting ideas to implement the toolkit, due to the heterogeneity of the circumstance.

6.1.2 *Open Questions*

The OP represent a varying panorama of educational experiences, with their own peculiar characteristics, and often not overlapping between them or attributable to an actual standard.

They are also designed and implemented by designers from different backgrounds, both scientific and operational, a factor which is reflected in OP quality, reproducibility and dissemination.

The assessment of learning outcomes and the impact determined by single initiatives are other open issues, because they are linked to the uncommon and extremely variable duration of the programs and the lack of longitudinal sessions on which to carry out the surveys.

As for the toolkit, it is certainly possible - and desirable - that it could be implemented, possibly after a testing phase and evaluation. The criteria adopted are entirely the result of my personal elaboration of the survey I produced but, during the M.O.D.O. design, I saw the possibility of further and different interpretations.

According to the principles and impact of Open Science, through the Outreach Toolkit I also encourage the collaboration within the community of researchers and practitioners. For this reason I adopted a Creative Common License 4.0 Share-alike to make it openly available.

6.2 SELF-DEVELOPMENT AND PROFESSIONAL ADVANCEMENT CONTRIBUTIONS

As a further conclusion of this Ph.D. research work, and also with respect to my initial perspectives, I would like to add that my doctoral path has represented to me a considerable possibility of personal enrichment on already elective topics.

In my educational and professional pipeline, I always considered this experience as an opportunity for development and training for my role as an educator and teacher.

I wanted to deepen many aspects of Computing Education as it is addressed in the school, but also to combine it with my parallel experience in the promotion of programming among children and teenagers in extracurricular environments.

In the Department of Computer Science at University of Salerno and in the research lab ISISLab where I performed my course, together with my tutor and colleagues, I found the way to undertake this path of awareness, as well as a specific and high-quality training in CS research and CEd.

It hasn't been always an easy journey, and I often questioned myself until the end of the road that brought me here. I struggled with the imposter syndrome always trying to take over, as I came from a completely different discipline and field. My non-IT background prompted me to a very in-depth study and opened up to my eyes further perspectives of investigation, from an interdisciplinary point of view.

The set of training activities carried out at the Department of Computer Science, the research experience made, the participation undertaken and those already in progress previously, gave me new IT knowledge and skills that I will bring back with me in my professional life, both as a teacher and a member of the public administration, in my job and continuing education.

[1] Agnese Addone and Luigi De Bernardis. "Chesscards: Making a Paper Chess Game with Primary School Students, a Cooperative Approach." In: *Makers at School, Educational Robotics and Innovative Learning Environments. Research and Experiences from FabLearn Italy 2019. Lecture Notes in Networks and Systems*. Vol. 240. Springer, 2021, pp. 141–146. DOI: 10.1007/978-3-030-77040-2_19.

[2] Agnese Addone, Renato De Donato, Giuseppina Palmieri, Maria Angela Pellegrino, Andrea Petta, Vittorio Scarano, and Luigi Serra. "Visual Storytelling by Novelette." In: *24th International Conference Information Visualisation (IV)*. IEEE, 2020, pp. 723–728. DOI: 10.1109/IV51561.2020.00126.

[3] Agnese Addone, Renato De Donato, Giuseppina Palmieri, Maria Angela Pellegrino, Andrea Petta, Vittorio Scarano, and Luigi Serra. "Novelette, a Usable Visual Storytelling Digital Learning Environment." In: *IEEE Access* 9 (2021), pp. 168850–168868. DOI: 10.1109/ACCESS.2021.3137076.

[4] Agnese Addone, Giuseppina Palmieri, and Maria Angela Pellegrino. "Engaging Children in Digital Storytelling." In: *Methodologies and intelligent Systems for Technology Enhanced Learning, 11th International Conference. MIS4TEL 2021. Lecture Notes in Networks and Systems*. Vol. 326. Springer, 2021, pp. 261–270. DOI: 10.1007/978-3-030-86618-1_26.

[5] Agnese Addone and Vittorio Scarano. "Outreach in K-12 Programming: A Systematic Literature Review on Audience and Purpose." In: *Proceedings of the 14th International Conference on Computer Supported Education, CSEDU, Volume 2*. SCITEPRESS, 2022, pp. 276–283. DOI: 10.5220/0011063400003182.

[6] Agnese Addone, Vittorio Scarano, and Carmine Spagnuolo. "Towards a collaborative taxonomy of Tools, Languages and Environments in K-12 Computing Education."

In: *IEEE Frontiers in Education Conference (FIE)*. 2022, pp. 1–5. DOI: 10.1109/FIE56618.2022.9962683.

[7] Carmelo Presicce and Agnese Addone. "L'apprendimento creativo dell'informatica in contesti informali: l'esperienza di CoderDojo." In: *Atti DidaMatica* (2014). Last accessed 2023, pp. 797–800. URL: https://mondodigitale.aicanet.net/2014-3/03_Computational_Thinking/index.xml.

[8] Fernando De la Prieta, Rosella Gennari, Marco Temperini, Tania Di Mascio, Pierpaolo Vittorini, Zuzana Kubincova, Elvira Popescu, Davide Rua Carneiro, Loreto Lancia, and Agnese Addone, eds. *Methodologies and Intelligent Systems for Technology Enhanced Learning, 11th International Conference. Lecture Notes in Networks and Systems*. Vol. 326. Springer, 2022. DOI: 10.1007/978-3-030-86618-1.

[1] Agnese Addone and Luigi De Bernardis. "Chesscards: Making a Paper Chess Game with Primary School Students, a Cooperative Approach." In: *Makers at School, Educational Robotics and Innovative Learning Environments. Research and Experiences from FabLearn Italy 2019. Lecture Notes in Networks and Systems*. Vol. 240. Springer, 2021, pp. 141–146. DOI: 10.1007/978-3-030-77040-2_19.

[2] Agnese Addone, Renato De Donato, Giuseppina Palmieri, Maria Angela Pellegrino, Andrea Petta, Vittorio Scarano, and Luigi Serra. "Visual Storytelling by Novelette." In: *24th International Conference Information Visualisation (IV)*. IEEE, 2020, pp. 723–728. DOI: 10.1109/IV51561.2020.00126.

[3] Agnese Addone, Renato De Donato, Giuseppina Palmieri, Maria Angela Pellegrino, Andrea Petta, Vittorio Scarano, and Luigi Serra. "Novelette, a Usable Visual Storytelling Digital Learning Environment." In: *IEEE Access* 9 (2021), pp. 168850–168868. DOI: 10.1109/ACCESS.2021.3137076.

[4] Agnese Addone, Giuseppina Palmieri, and Maria Angela Pellegrino. "Engaging Children in Digital Storytelling." In: *Methodologies and intelligent Systems for Technology Enhanced Learning, 11th International Conference. MIS4TEL 2021. Lecture Notes in Networks and Systems*. Vol. 326. Springer, 2021, pp. 261–270. DOI: 10.1007/978-3-030-86618-1_26.

[5] Agnese Addone and Vittorio Scarano. "Outreach in K-12 Programming: A Systematic Literature Review on Audience and Purpose." In: *Proceedings of the 14th International Conference on Computer Supported Education, CSEDU, Volume 2*. SCITEPRESS, 2022, pp. 276–283. DOI: 10.5220/0011063400003182.

[6] Agnese Addone, Vittorio Scarano, and Carmine Spagnuolo. "Towards a collaborative taxonomy of Tools, Languages and Environments in K-12 Computing Education."

In: *IEEE Frontiers in Education Conference (FIE)*. 2022, pp. 1–5. DOI: 10.1109/FIE56618.2022.9962683.

[7] Abeer Alsheaibi, Glenn Strong, and Richard Millwood. "The Need for a Learning Model in Coderdojo Mentoring Practice." In: *Proceedings of the 13th Workshop in Primary and Secondary Computing Education*. WiPSCE '18. Potsdam, Germany: Association for Computing Machinery, 2018. DOI: 10.1145/3265757.3265785.

[8] Mikko Apiola, Mohammed Saqr, Sonsoles López-Pernas, and Matti Tedre. "Computing Education Research Compiled: Keyword Trends, Building Blocks, Creators, and Dissemination." In: *IEEE Access* 10 (2022), pp. 27041–27068. DOI: 10.1109/ACCESS.2022.3157609.

[9] Karina Assiter and Charlie Wiseman. "Exploratory Learning with Alice: Experiences Leading a Computer Science Workshop for Girl Scouts." In: *Journal of Computing Sciences in Colleges* 31.4 (2016), 21–27. ISSN: 1937-4771. DOI: 10.5555/2904127.2904130.

[10] Tiffany Barnes. "CS for All, Equity, and Responsibility." In: *SIGCSE Bull.* 49.2 (2017), p. 18. ISSN: 0097-8418. DOI: 10.1145/3094875.3094882.

[11] Brett A. Becker and Keith Quille. "50 Years of CS1 at SIGCSE: A Review of the Evolution of Introductory Programming Education Research." In: *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*. SIGCSE '19. Association for Computing Machinery, 2019, 338–344. DOI: 10.1145/3287324.3287432.

[12] Andrew Begel and Amy J. Ko. "Learning Outside the Classroom." In: *The Cambridge Handbook of Computing Education Research*. Ed. by Sally A. Fincher and Anthony V. Robins. Cambridge Handbooks in Psychology. Cambridge University Press, 2019, 749–772. DOI: 10.1017/9781108654555.027.

[13] Scott Bell, Michael Rogers, Diana Linville, and Cari Cline. "Building a K-12 Community of Practice." In: *Journal of Computing Sciences in Colleges* 34.2 (2018), 12–18. ISSN: 1937-4771. DOI: 10.5555/3282588.3282590.

[14] Tim Bell, Peter Andreae, and Anthony Robins. "A Case Study of the Introduction of Computer Science in NZ Schools." In: *ACM Transactions on Computing Education* 14.2 (2014). DOI: 10.1145/2602485.

[15] Miles Berry. *Computing in the national curriculum: A guide for primary teachers*. British Computer Society, 2013. ISBN: 978-1-78339-143-1.

[16] Paulo Blikstein and Sepi Hejazi Moghadam. *Pre-college computer science education: A survey of the field*. 2018. URL: https://research.google/pubs/pub49719/.

[17] Paulo Blikstein and Sepi Hejazi Moghadam. "Computing education." In: *The Cambridge handbook of computing education research*. Cambridge University Press, 2019, pp. 56–78. DOI: 10.1017/9781108654555.004.

[18] Benjamin S. Bloom et al. *Taxonomy of educational objectives: The classification of educational goals*. McKay Company, 1956. ISBN: 978-0679302117.

[19] Danah Boyd. *It's complicated: The social lives of networked teens*. Yale University Press, 2014. ISBN: 9780300166316.

[20] Karen Brennan. "A case for why: Society, school, self." In: *Computational Thinking Education in K–12: Artificial Intelligence Literacy and Physical Computing*. The MIT Press, 2022. DOI: 10.7551/mitpress/13375.003.0004.

[21] Ricardo Britto and Muhammad Usman. "Bloom's Taxonomy in Software Engineering Education: A Systematic Mapping Study." In: *Proceedings of the 2015 IEEE Frontiers in Education Conference (FIE)*. FIE '15. USA: IEEE Computer Society, 2015, 1–8. DOI: 10.1109/FIE.2015.7344084.

[22] Jerome Bruner. *Acts of meaning: Four Lectures on Mind and Culture*. Harvard University Press, 1990. ISBN: 9780674003613.

[23] CS4All. *CS4All*. Last accessed 2023. URL: https://www.csforall.org/.

[24] Tracy Camp, Christine Liebe, and Michelle Slattery. "Applying NCWIT Protocol to Broaden Participation in Computing: A Case Study of CS@Mines." In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. New York, NY, USA: Association for Computing Machinery, 2020, 528–534. DOI: 10.1145/3328778.3366958.

[25] Veronica Cateté, Nicholas Lytle, and Tiffany Barnes. "Creation and Validation of Low-Stakes Rubrics for K-12 Computer Science." In: *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE 2018. Larnaca, Cyprus: Association for Computing Machinery, 2018, 63–68. DOI: 10.1145/3197091.3197134.

[26] Chitat Chan and Carmen Yau. "Digital storytelling for social work interventions." In: *Social work* (2019). DOI: 10.1093/OBO/9780195389678-0273.

[27] Hannah E. Chipman, Fernando J. Rodríguez, and Kristy Elizabeth Boyer. ""I Impressed Myself With How Confident I Felt": Reflections on a Computer Science Assessment for K-8 Teachers." In: *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*. SIGCSE '19. Minneapolis, MN, USA: Association for Computing Machinery, 2019, 1081–1087. DOI: 10.1145/3287324.3287478.

[28] Hannah Chipman, Haley Adams, Betsy Williams Sanders, and D. Brian Larkins. "Evaluating Computer Science Camp Topics in Increasing Girls' Confidence in Computer Science." In: *J. Comput. Sci. Coll.* 33.5 (2018), 70–78. ISSN: 1937-4771. DOI: 10.5555/3204979.3204993.

[29] European CodeWeek. *European CodeWeek*. Last accessed 2023. URL: https://codeweek.eu/.

[30] Code.org. *Code.org*. Last accessed 2023. URL: https://code.org/.

[31] CoderDojo. *CoderDojo*. Last accessed 2023. URL: https://coderdojo.com/en/.

[32] Merijke Coenraad, Bih Janet Fofang, and David Weintrop. "Gusanos y Espheros: Computing with Youth in Rural El Salvador." In: *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*. Association for Computing Machinery, 2021, 404–410. DOI: 10.1145/3408877.3432535.

[33] European Commission, European Education, and Culture Executive Agency. *Informatics Education at School in Europe*. 2022. DOI: 10.2797/268406.

[34] Sasha Costanza-Chock. *Design justice: Community-led practices to build the worlds we need*. The MIT Press, 2020. ISBN: 9780262043458.

[35] Tyne Crow, Andrew Luxton-Reilly, Burkhard C. Wünsche, and Paul Denny. "Resources and Support for the Implementation of Digital Technologies in New Zealand Schools." In: *Proceedings of the Twenty-First Australasian Computing Education Conference*. ACE '19. Sydney, NSW, Australia: Association for Computing Machinery, 2019, 69–78. DOI: 10.1145/3286960.3286969.

[36] Robert Cummings, Brittany Chambers, Amber Reid, and Kinnis Gosha. "STEM Hip-Hop Pedagogy: A Meta-Synthesis on Hip-Hop Pedagogy STEM Interventions Tools for Underrepresented Minorities in K-12 Education." In: *Proceedings of the 2019 ACM Southeast Conference*. ACM SE '19. Association for Computing Machinery, 2019, 46–52. DOI: 10.1145/3299815.3314431.

[37] Valentina Dagienė, Gabrielė Stupurienė, and Lina Vinikienė. "Promoting Inclusive Informatics Education Through the Bebras Challenge to All K-12 Students." In: *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016*. CompSysTech '16. Palermo, Italy: Association for Computing Machinery, 2016, 407–414. DOI: 10.1145/2983468.2983517.

[38] Anita DeWitt et al. "Arts Coding for Social Good: A Pilot Project for Middle-School Outreach." In: *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. SIGCSE '17. Seattle, Washington, USA:

Association for Computing Machinery, 2017, 159–164. DOI: 10.1145/3017680.3017795.

[39] Adrienne Decker and Monica M. McGill. "Pre-College Computing Outreach Research: Towards Improving the Practice." In: *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. SIGCSE '17. Seattle, Washington, USA: Association for Computing Machinery, 2017, 153–158. DOI: 10.1145/3017680.3017744.

[40] Adrienne Decker, Monica M. McGill, and Amber Settle. "Towards a Common Framework for Evaluating Computing Outreach Activities." In: *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*. SIGCSE '16. Association for Computing Machinery, 2016, 627–632. DOI: 10.1145/2839509.2844567.

[41] Katrina Falkner, Rebecca Vivian, and Nickolas Falkner. "The Australian Digital Technologies Curriculum: Challenge and Opportunity." In: *Proceedings of the Sixteenth Australasian Computing Education Conference - Volume 148*. ACE '14. Australian Computer Society, Inc., 2014, 3–12. DOI: 10.5555/2667490.2667491.

[42] Brittany Terese Fasy, Stacey A. Hancock, Barbara Z. Komlos, Brendan Kristiansen, Samuel Micka, and Allison S. Theobold. "Bring the Page to Life: Engaging Rural Students in Computer Science Using Alice." In: *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE '20. Trondheim, Norway: Association for Computing Machinery, 2020, 110–116. DOI: 10.1145/3341525.3387367.

[43] Matthias Felleisen, Robert Bruce Findler, Matthew Flatt, and Shriram Krishnamurthi. *How to design programs: an introduction to programming and computing*. 2018. ISBN: 978-0262534802.

[44] W. Feurzeig, S. Papert, M. Bloom, R. Grant, and C. Solomon. "Programming-Languages as a Conceptual Framework for Teaching Mathematics." In: *SIGCUE Outlook* 4.2 (1970), 13–17. DOI: 10.1145/965754.965757.

[45] Sally A Fincher and Anthony V Robins. *The Cambridge handbook of computing education research*. Cambridge University Press, 2019. ISBN: 978-1108721899.

[46] Cynthia J. Finelli, Maura Borrego, and Golnoosh Rasoulifar. "Development of a Taxonomy of Keywords for Engineering Education Research." In: *Journal of Engineering Education* 104.4 (2015), pp. 365–387. DOI: https://doi.org/10.1002/jee.20101.

[47] Adam Fletcher, Raina Mason, and Graham Cooper. "Helping Students Get IT: Investigating the Longitudinal Impacts of IT School Outreach in Australia." In: *Proceedings of the 23rd Australasian Computing Education Conference*. ACE '21. Virtual, SA, Australia: Association for Computing Machinery, 2021, 115–124. DOI: 10.1145/3441636.3442312.

[48] Dan Garcia, Moses Charikar, Eboney Hearn, Ed Lazowska, and Jonathan Reynolds. "Institutions Share Successes, Failures, and Advice in Moving the Diversity Needle." In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. New York, NY, USA: Association for Computing Machinery, 2020, 331–332. DOI: 10.1145/3328778.3366976.

[49] Tracy Gardner, Hayley C. Leonard, Jane Waite, and Sue Sentance. "What Do We Know about Computing Education for K-12 in Non-Formal Settings? A Systematic Literature Review of Recent Research." In: *Proceedings of the 2022 ACM Conference on International Computing Education Research*. Vol. 1. ICER '22. Association for Computing Machinery, 2022, 264–281. DOI: 10.1145/3501385.3543960.

[50] James Geller and Soon Ae Chun. "Inclusive Education Strategies for Diversity in Smart Workforce." In: *DG.O2021: The 22nd Annual International Conference on Digital Government Research*. DG.O'21. Association for Computing Machinery, 2021, 264–272. DOI: 10.1145/3463677.3463741.

[51] Henry Giroux. *On Critical pedagogy*. Bloomsbury, 2020. ISBN: 9781350144989.

[52] GitHub. *GitHub*. https://github.com. Last accessed 2022.

[53] GitHubPages. *GitHubPages*. https://pages.github.com/. Last accessed 2022.

[54] Google. *GoogleSheet*. https://docs.google.com/spreadsheets. Last accessed 2022.

[55] Martin Greenberger. *Computers and the World of the Future*. MIT Press, 1964. ISBN: 9780262570022.

[56] Shuchi Grover. *Computer Science in K-12: An A-Z Handbook on Teaching Programing*. June 2020. ISBN: 9781734662702.

[57] Mark Guzdial and Benedict du Boulay. "The History of Computing." In: *The Cambridge Handbook of Computing Education Research (2019)*. Ed. by Sally A. Fincher and Anthony V. Robins. Vol. 11. 2019. DOI: 10.1017/9781108654555.002.

[58] Mark Guzdial, Emma Dodoo, Bahare Naimpour, Tamara Nelson-Fromm, and Aadarsh Padiyath. "Putting a Teaspoon of Programming into Other Subjects." In: *Commun. ACM* 66.5 (2023), 24–26. DOI: 10.1145/3587926.

[59] Margaret Hamilton et al. "Gender Equity in Computing: International Faculty Perceptions and Current Practices." In: *Proceedings of the 2016 ITiCSE Working Group Reports*. ITiCSE '16. Association for Computing Machinery, 2016, 81–102. DOI: 10.1145/3024906.3024911.

[60] Lelia Hampton, Robert Cummings, and Kinnis Gosha. "Improving Computer Science Instruction and Computer Use for African American Secondary School Students: A Focus Group Exploration of Computer Science Identity of African American Teachers." In: *Proceedings of the 2019 on Computers and People Research Conference*. SIGMIS-CPR '19. Nashville, TN, USA: Association for Computing Machinery, 2019, 78–84. DOI: 10.1145/3322385.3322399.

[61] Brian Harvey, Daniel Garcia, Josh Paley, and Luke Segars. "Snap! (Build Your Own Blocks) (Abstract Only)." In: *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*. SIGCSE '12. Raleigh, North Carolina, USA: Association for Computing Machinery, 2012, p. 662. DOI: 10.1145/2157136.2157351.

[62] Pekka Himanen. *The hacker ethic*. Random House, 2002. ISBN: 9780375758782.

[63] Michael Stephen Horn and Marina U. Bers. "Tangible Computing." In: *The Cambridge Handbook of Computing Education Research*. Ed. by Sally A. Fincher and Anthony V. Robins. Cambridge University Press, 2019, pp. 663–678. DOI: 10.1017/9781108654555.023.

[64] Mizuko Ito, Kris Gutiérrez, Sonia Livingstone, Bill Penuel, Jean Rhodes, Katie Salen, Juliet Schor, Julian Sefton-Green, and S Craig Watkins. *Connected learning: An agenda for research and design*. Digital Media and Learning Research Hub, 2013. ISBN: 978-0-9887255-0-8.

[65] Letizia Jaccheri, Cristina Pereira, and Swetlana Fast. "Gender Issues in Computer Science: Lessons Learnt and Reflections for the Future." In: *2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*. IEEE. 2020, pp. 9–16. DOI: 10.1109/SYNASC51798.2020.00014.

[66] Gayithri Jayathirtha, Joanna Goode, Nicki Washington, Shana V. White, Aman Yadav, and Cecilé Sadler. "Supporting the Integration of Social Justice Topics within K-12 Computing Education." In: *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 2*. SIGCSE 2023. Toronto ON, Canada: Association for Computing Machinery, 2023, 1206–1207. DOI: 10.1145/3545947.3569604.

[67] Jekyll. *Jekyll*. https://jekyllrb.com/. Last accessed 2022.

[68] Wei Jin, Xin Xu, and Sonal Dekhane. "Light-Weight Student-Driven Workshops for Positive Attitude Change towards Programming in Early College." In: *J. Comput. Sci. Coll.* 35.7 (2020), 61–73. ISSN: 1937-4771. DOI: 10.5555/3417626.3417632.

[69] Simon Peyton Jones, Tim Bell, Quintin Cutts, Sridhar Iyer, Carsten Schulte, Jan Vahrenhold, and ByoungRae Han. "Computing at school." In: *International comparisons* (2011).

[70] Eunjin (EJ) Jung and Xornam S. Apedoe. "Changing Young Women's Perceptions of CS via Outreach." In: *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE '13. Canterbury, England, UK: Association for Computing Machinery, 2013, p. 333. DOI: 10.1145/2462476.2465616.

[71] Yasmin B Kafai and Quinn Burke. *Connected code: Why children need to learn programming*. Mit Press, 2014. DOI: 10.7551/mitpress/9992.001.0001.

[72] Kristy Kallback-Rose, Kurt Seiffert, Danko Antolovic, Therese Miller, Robert Ping, and Craig Stewart. "Conducting K-12 Outreach to Evoke Early Interest in IT, Science, and Advanced Technology." In: *Proceedings of the 1st Conference of the Extreme Science and Engineering Discovery Environment: Bridging from the EXtreme to the Campus and Beyond*. XSEDE '12. Chicago, Illinois, USA: Association for Computing Machinery, 2012. DOI: 10.1145/2335755.2335853.

[73] Peerapon Kamlangpuech and Komate Amphawan. "A new system for analyzing contents of Computer Science courses." In: *2020 7th International Conference on Advance Informatics: Concepts, Theory and Applications (ICAICTA)*. 2020, pp. 1–6. DOI: 10.1109/ICAICTA49861.2020.9429072.

[74] Alan C. Kay. "A Personal Computer for Children of All Ages." In: *Proceedings of the ACM Annual Conference - Volume 1*. Boston, Massachusetts, USA: Association for Computing Machinery, 1972. DOI: 10.1145/800193.1971922.

[75] Peter E. J. Kemp, Billy Wong, and Miles G. Berry. "Female Performance and Participation in Computer Science: A National Picture." In: *ACM Trans. Comput. Educ.* 20.1 (2019). DOI: 10.1145/3366016.

[76] Barbara Kitchenham. "Procedures for performing systematic reviews." In: *Keele, UK, Keele University* 33.2004 (2004), pp. 1–26.

[77] Barbara Kitchenham, Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen Linkman. "Systematic literature reviews in software engineering. A systematic literature review." In: *Information and software technology*

51.1 (2009), pp. 7–15. DOI: 10.1016/j.infsof.2008.09.009.

[78] David R Krathwohl. "A revision of Bloom's taxonomy: An overview." In: *Theory into practice* 41.4 (2002), pp. 212–218. DOI: 10.1207/s15430421tip4104_2.

[79] Shriram Krishnamurthi, Emmanuel Schanzer, Joe Gibbs Politz, Benjamin S. Lerner, Kathi Fisler, and Sam Dooman. *Data Science as a Route to AI for Middle- and High-School Students*. 2020. arXiv: 2005.01794 [cs.CY].

[80] Consorzio CINI Laboratorio Informatica & Scuola. *Consorzio CINI, Laboratorio Informatica & Scuola*. Last accessed 2023. URL: https://www.consorzio-cini.it/index.php/it/gdl-informatica-scuola.

[81] Richard Ladner and Tammy VanDeGrift. "Special Issue on Broadening Participation in Computing Education (Part 2)." In: *ACM Trans. Comput. Educ.* 11.3 (2011). DOI: 10.1145/2037276.2037277.

[82] Adidah Lajis, Haidawati Md Nasir, and Normaziah A. Aziz. "Proposed Assessment Framework Based on Bloom Taxonomy Cognitive Competency: Introduction to Programming." In: *Proceedings of the 2018 7th International Conference on Software and Computer Applications*. ICSCA 2018. Kuantan, Malaysia: Association for Computing Machinery, 2018, 97–101. ISBN: 9781450354141. DOI: 10.1145/3185089.3185149.

[83] Antti-Jussi Lakanen and Tommi Kärkkäinen. "Identifying Pathways to Computer Science: The Long-Term Impact of Short-Term Game Programming Outreach Interventions." In: *ACM Trans. Comput. Educ.* 19.3 (2019). DOI: 10.1145/3283070.

[84] Catherine Lang, Annemieke Craig, and MaryAnne Egan. "The Importance of Outreach Programs to Unblock the Pipeline and Broaden Diversity in ICT Education." In: *Int. J. Inf. Commun. Technol. Educ.* 12.1 (2016), 38–49. ISSN: 1550-1876. DOI: 10.4018/IJICTE.2016010104.

[85] Grace Lawlor, Philip Byrne, and Brendan Tangney. ""Code-Plus"—Measuring Short-Term Efficacy in a Non-Formal, All-Female CS Outreach Programme." In: *ACM Trans. Comput. Educ.* 20.4 (2020). DOI: 10.1145/3411510.

[86] Jay Lemke, Robert Lecusay, Michael Cole, and Vera Michalchik. *Documenting and assessing learning in informal and media-rich environments*. The MIT Press, 2015. ISBN: 9780262527743.

[87] Sylvia Libow Martinez and Gary Stager. *Invent to learn: Making, Tinkering and Engineering in the Classroom*. Constructing Modern Knowledge, 2013. ISBN: 978-0989151108.

[88] Yuhan Lin and David Weintrop. "The landscape of Block-based programming: Characteristics of block-based environments and how they support the transition to text-based programming." In: *Journal of Computer Languages* 67 (2021), p. 101075. DOI: 10.1016/j.cola.2021.101075.

[89] Stephanie Ludi, George Adams, Bradley Blankenship, and Michael Dapiran. "The Architectural Challenges of Adding Accessibility Features to ALICE as a Case Study of Maintenance in Educational Software." In: GAS '11. Honolulu, HI, USA: Association for Computing Machinery, 2011, 33–35. ISBN: 9781450305785. DOI: 10.1145/1984674.1984686.

[90] Stephanie Ludi, Lindsey Ellis, and Scott Jordan. "An Accessible Robotics Programming Environment for Visually Impaired Users." In: ASSETS '14. Rochester, New York, USA: Association for Computing Machinery, 2014, 237–238. ISBN: 9781450327206. DOI: 10.1145/2661334.2661385.

[91] Stephanie Ludi and Tom Reichlmayr. "The Use of Robotics to Promote Computing to Pre-College Students with Visual Impairments." In: *ACM Trans. Comput. Educ.* 11.3 (2011). DOI: 10.1145/2037276.2037284.

[92] Andrew Luxton-Reilly et al. "Introductory Programming: A Systematic Literature Review." In: *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE 2018

Companion. Association for Computing Machinery, 2018, 55–106. DOI: 10.1145/3293881.3295779.

[93] MakerFaireRome. *MakerFaireRome*. Last accessed 2023. URL: https://makerfairerome.eu/it/.

[94] Lauri Malmi. "Reflections on Theory." In: *ACM Transactions on Computing Education* 23.1 (2022). DOI: 10.1145/3570728.

[95] Lauri Malmi, Judy Sheard, Päivi Kinnunen, Simon, and Jane Sinclair. "Computing Education Theories: What Are They and How Are They Used?" In: *ACM Inroads* 11.1 (2020), 54–64. ISSN: 2153-2184. DOI: 10.1145/3381889.

[96] Lauri Malmi, Judy Sheard, Päivi Kinnunen, Simon, and Jane Sinclair. "Theories and Models of Emotions, Attitudes, and Self-Efficacy in the Context of Programming Education." In: *Proceedings of the 2020 ACM Conference on International Computing Education Research*. ICER '20. Virtual Event, New Zealand: Association for Computing Machinery, 2020, 36–47. ISBN: 9781450370929. DOI: 10.1145/3372782.3406279.

[97] Lauri Malmi, Ian Utting, and Amy J. Ko. "Tools and Environments." In: *The Cambridge Handbook of Computing Education Research*. Ed. by Sally A. Fincher and Anthony V.Editors Robins. Cambridge Handbooks in Psychology. Cambridge University Press, 2019, 639–662. DOI: 10.1017/9781108654555.022.

[98] John Maloney, Mitchel Resnick, Natalie Rusk, Brian Silverman, and Evelyn Eastmond. "The Scratch Programming Language and Environment." In: *ACM Trans. Comput. Educ.* 10.4 (2010). DOI: 10.1145/1868358.1868363.

[99] Victoria J Marsick and Karen E Watkins. "Informal and incidental learning." In: *New directions for adult and continuing education* 2001.89 (2001), pp. 25–34. DOI: https://doi.org/10.1002/ace.5.

[100] Susana Masapanta-Carrión and J. Ángel Velázquez-Iturbide. "A Systematic Review of the Use of Bloom's Taxonomy in Computer Science Education." In: *Proceedings of the*

49th ACM Technical Symposium on Computer Science Education. SIGCSE '18. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, 441–446. DOI: 10.1145/3159450.3159491.

[101] Andrew McDonald and Laura K. Dillon. "Virtual Outreach: Lessons from a Coding Club's Response to COVID-19." In: *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*. SIGCSE '21. New York, NY, USA: Association for Computing Machinery, 2021, 418–424. DOI: 10.1145/3408877.3432559.

[102] Monica M. McGill and Adrienne Decker. "Construction of a Taxonomy for Tools, Languages, and Environments across Computing Education." In: *Proceedings of the 2020 ACM Conference on International Computing Education Research*. ICER '20. Association for Computing Machinery, 2020, 124–135. DOI: 10.1145/3372782.3406258.

[103] Monica M. McGill and Adrienne Decker. "Tools, Languages, and Environments Used in Primary and Secondary Computing Education." In: ITiCSE '20. Trondheim, Norway: Association for Computing Machinery, 2020, 103–109. ISBN: 9781450368742. DOI: 10.1145/3341525.3387365.

[104] Monica M. McGill, Adrienne Decker, and Amber Settle. "Undergraduate Students' Perceptions of the Impact of Pre-College Computing Activities on Choices of Major." In: *ACM Trans. Comput. Educ.* 16.4 (2016). DOI: 10.1145/2920214.

[105] MediaWiki. *MediaWiki*. https://www.mediawiki.org/. Last accessed 2022.

[106] Andreea Molnar, Therese Keane, and Rosemary Stockdale. "Educational Interventions and Female Enrollment in IT Degrees." In: *Commun. ACM* 64.3 (2021), 73–77. ISSN: 0001-0782. DOI: 10.1145/3387106.

[107] Hamzeh Moradi and Hefang Chen. ""Digital Storytelling in Language Education"." In: *Behavioral Sciences* vol. 9.n. 12 (2019), p. 147. DOI: https://doi.org/10.3390/bs9120147.

[108] Jesús Moreno-León and Gregorio Robles. "The Europe code week (CodeEU) initiative shaping the skills of future engineers." In: *2015 IEEE global engineering education conference (EDUCON)*. IEEE. 2015, pp. 561–566. DOI: 10.1109/EDUCON.2015.7096025.

[109] Greg L. Nelson and Amy J. Ko. "On Use of Theory in Computing Education Research." In: *Proceedings of the 2018 ACM Conference on International Computing Education Research*. ICER '18. Espoo, Finland: Association for Computing Machinery, 2018, 31–39. DOI: 10.1145/3230977.3230992.

[110] Karen Nolan, Roisin Faherty, Keith Quille, Brett A. Becker, and Susan Bergin. "Developing an Inclusive K-12 Outreach Model." In: *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE '20. Trondheim, Norway: Association for Computing Machinery, 2020, 145–151. DOI: 10.1145/3341525.3387421.

[111] Notion. *Notion*. https://www.notion.so. Last accessed 2022.

[112] Elizabeth Oldham. "Computing in the Irish School Curriculum: What Can We Learn from a Fifty-Year Adventure?" In: *Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education Vol. 1*. ITiCSE '22. Dublin, Ireland: Association for Computing Machinery, 2022, p. 1. DOI: 10.1145/3502718.3524817.

[113] Alannah Oleson, Brett Wortzman, and Amy J. Ko. "On the Role of Design in K-12 Computing Education." In: *ACM Trans. Comput. Educ.* 21.1 (2021). DOI: 10.1145/3427594.

[114] OpenEducationItalia. *OpenEducationItalia*. Last accessed 2023. URL: https://openeducationitalia.it/.

[115] Seymour Papert. *Mindstorms: Children, Computers, and Powerful Ideas*. USA: Basic Books, Inc., 1980. ISBN: 0465046274.

[116] Seymour Papert. *The children's machine: Rethinking school in the age of the computer*. Basic Books, Inc., 1993. ISBN: 0465018300.

[117] Hadi Partovi. "Transforming US Education with Computer Science." In: *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*. SIGCSE '14. Atlanta, Georgia, USA: Association for Computing Machinery, 2014, 5–6. DOI: 10.1145/2538862.2554793.

[118] Joslenne Pena. "Seeding the Computational Skills of Diverse Non-Programmers through Non-Formal Workshops." In: *Proceedings of the 2019 ACM Conference on International Computing Education Research*. ICER '19. Toronto ON, Canada: Association for Computing Machinery, 2019, 347–348. ISBN: 9781450361859. DOI: 10.1145/3291279.3339444.

[119] Jeffrey Perkel. "Democratic Databases: Science on GitHub." In: *Nature* 538.7623 (2016), pp. 127–128. DOI: 10.1038/538127a.

[120] Radia Perlman. *Using computer technology to provide a creative learning environment for preschool children*. Last accessed 2023. 1976. URL: https://dspace.mit.edu/handle/1721.1/5784.

[121] Marc Prensky. "Digital natives, digital immigrants. Part 1." In: *On the Horizon* 9.5 (2001), pp. 1–6. DOI: https://doi.org/10.1108/10748120110424816.

[122] Carmelo Presicce and Agnese Addone. "L'apprendimento creativo dell'informatica in contesti informali: l'esperienza di CoderDojo." In: *Atti DidaMatica* (2014). Last accessed 2023, pp. 797–800. URL: https://mondodigitale.aicanet.net/2014-3/03_Computational_Thinking/index.xml.

[123] Fernando De la Prieta, Rosella Gennari, Marco Temperini, Tania Di Mascio, Pierpaolo Vittorini, Zuzana Kubincova, Elvira Popescu, Davide Rua Carneiro, Loreto Lancia, and Agnese Addone, eds. *Methodologies and Intelligent Systems for Technology Enhanced Learning, 11th International Conference. Lecture Notes in Networks and Systems*. Vol. 326. Springer, 2022. DOI: 10.1007/978-3-030-86618-1.

[124] Coolest Projects. *Coolest Projects*. Last accessed 2023. URL: https://online.coolestprojects.org/.

[125] Pooja Rana, Lovi Raj Gupta, Gulshan Kumar, and Mithilesh Kumar Dubey. “A Taxonomy of Various Applications of Artificial Intelligence in Education.” In: *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)*. 2021, pp. 23–28. DOI: 10.1109/ICIEM51511.2021.9445339.

[126] Mitchel Resnick, Stephen Ocko, and Seymour Papert. “LEGO, Logo, and Design.” In: *Children’s Environments Quarterly* 5.4 (1988), pp. 14–18. URL: http://www.jstor.org/stable/41514692 (visited on 2023).

[127] Mitchel Resnick et al. “Scratch: Programming for All.” In: *Commun. ACM* 52.11 (2009), 60–67. DOI: 10.1145/1592761.1592779.

[128] Penny Rheingans, Erica D’Eramo, Crystal Diaz-Espinoza, and Danyelle Ireland. “A Model for Increasing Gender Diversity in Technology.” In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. SIGCSE ’18. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, 459–464. DOI: 10.1145/3159450.3159533.

[129] Michele Roberts, Kiki Prottsman, and Jeff Gray. “Priming the Pump: Reflections on Training K-5 Teachers in Computer Science.” In: SIGCSE ’18. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, 723–728. DOI: 10.1145/3159450.3159560.

[130] Gianni Rodari. *Grammatica della fantasia. Introduzione all’arte di inventare storie*. Ed. by Einaudi. Torino, 1973. ISBN: 88-06-03817-6.

[131] RomeVideoGameLab. *RomeVideoGameLab*. Last accessed 2023. URL: https://romevideogamelab.it/it/.

[132] Ricarose Roque, Natalie Rusk, and Mitchel Resnick. “Supporting diverse and creative collaboration in the Scratch online community.” In: *Mass collaboration and education* (2016), pp. 241–256. DOI: 10.1007/978-3-319-13536-6_12.

[133] Douglas Rushkoff. *Program or be programmed: Ten commands for a digital age*. Or Books, 2010. ISBN: 978-1-935928-15-7.

[134] Mihaela Sabin, Paul Snow, and Mary Laturnau. "Evaluation of a Computing and Engineering Outreach Program for Girls in Grades 8–10." In: *J. Comput. Sci. Coll.* 30.6 (2015), 119–126. ISSN: 1937-4771. DOI: 10.5555/2753024.2753049.

[135] Daisuke Saito, Ayana Sasaki, Hironori Washizaki, Yoshiaki Fukazawa, and Yusuke Muto. "Program learning for beginners: Survey and taxonomy of programming learning tools." In: *2017 IEEE 9th International Conference on Engineering Education (ICEED)*. 2017, pp. 137–142. DOI: 10.1109/ICEED.2017.8251181.

[136] David Scaradozzi, Laura Screpanti, and Lorenzo Cesaretti. "Towards a Definition of Educational Robotics: A Classification of Tools, Experiences and Assessments." In: June 2019, pp. 63–92. DOI: 10.1007/978-3-030-19913-5_3.

[137] Emmanuel Schanzer, Shriram Krishnamurthi, and Kathi Fisler. "What Does It Mean for a Computing Curriculum to Succeed?" In: *Commun. ACM* 62.5 (2019), 30–32. ISSN: 0001-0782. DOI: 10.1145/3319081.

[138] Scratch. *Scratch*. Last accessed 2023. URL: https://scratch.mit.edu/.

[139] Linda Seiter. "Using SOLO to Classify the Programming Responses of Primary Grade Students." In: *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*. SIGCSE '15. Association for Computing Machinery, 2015, 540–545. DOI: 10.1145/2676723.2677244.

[140] Sue Sentance, Erik Barendsen, Nicol R. Howard, and Carsten Schulte. *Computer Science Education: Perspectives on Teaching and Learning in School*. Bloomsbury Publishing, 2023. DOI: 10.5040/9781350296947.

[141] Irene Sheridan, Deirdre Goggin, and Linda O'Sullivan. "Exploration of learning gained through coderdojo coding activities." In: *INTED 2016, the 10th International Technology, Education and Development Conference*. International

Academy of Technology, Education and Development, 2016. DOI: 10.21125/inted.2016.0545.

[142] Kristen Shinohara, Michael McQuaid, and Nayeri Jacobo. "Access Differential and Inequitable Access: Inaccessibility for Doctoral Students in Computing." In: *The 22nd International ACM SIGACCESS Conference on Computers and Accessibility*. ASSETS '20. Virtual Event, Greece: Association for Computing Machinery, 2020. ISBN: 9781450371032. DOI: 10.1145/3373625.3416989.

[143] Madhu Singh. "UNESCO GUIDELINES for the Recognition, Validation and Accreditation of the Outcomes of Non-formal and Informal Learning." In: *PLA Inside Out: An International Journal on Theory, Research and Practice in Prior Learning Assessment* 1.2 (2012). Last accessed 2023. URL: https://unesdoc.unesco.org/ark:/48223/pf0000216360.locale=en.

[144] Cynthia Solomon, Brian Harvey, Ken Kahn, Henry Lieberman, Mark L. Miller, Margaret Minsky, Artemis Papert, and Brian Silverman. "History of Logo." In: *Proc. ACM Program. Lang.* 4.HOPL (2020). DOI: 10.1145/3386329.

[145] La Scuola Open Source. *La Scuola Open Source*. Last accessed 2023. URL: https://www.lascuolaopensource.xyz/.

[146] Gary S Stager. "Seymour Papert (1928–2016)." In: *Nature* 537.7620 (2016), pp. 308–308. DOI: 10.1038/537308a.

[147] Jeffrey A. Stone. "Student Perceptions of Computing and Computing Majors." In: *J. Comput. Sci. Coll.* 34.3 (2019), 22–30. DOI: 10.5555/3306465.3306471.

[148] Claudia Szabo, Nickolas Falkner, Andrew Petersen, Heather Bort, Cornelia Connolly, Kathryn Cunningham, Peter Donaldson, Arto Hellas, James Robinson, and Judy Sheard. "A Periodic Table of Computing Education Learning Theories." In: *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE '19. Aberdeen, Scotland Uk: Association for Computing Machinery, 2019, 269–270. DOI: 10.1145/3304221.3325534.

[149] Claudia Szabo, Nickolas Falkner, Andrew Petersen, Heather Bort, Kathryn Cunningham, Peter Donaldson, Arto Hellas, James Robinson, and Judy Sheard. “Review and Use of Learning Theories within Computer Science Education Research: Primer for Researchers and Practitioners.” In: *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education*. ITiCSE-WGR ’19. Association for Computing Machinery, 2019, 89–109. DOI: 10.1145/3344429.3372504.

[150] Claudia Szabo and Judy Sheard. “Learning Theories Use and Relationships in Computing Education Research.” In: *ACM Trans. Comput. Educ.* 23.1 (2022). DOI: 10.1145/3487056.

[151] Claudia Szabo, Judy Sheard, Andrew Luxton-Reilly, Simon, Brett A. Becker, and Linda Ott. “Fifteen Years of Introductory Programming in Schools: A Global Overview of K-12 Initiatives.” In: *Proceedings of the 19th Koli Calling International Conference on Computing Education Research*. Koli Calling ’19. Association for Computing Machinery, 2019. DOI: 10.1145/3364510.3364513.

[152] Matti Tedre. “The Nature of Computing as a Discipline.” In: *Computer Science Education: Perspectives on Teaching and Learning in School*. Ed. by Sue Sentance, Erik Barendsen, Nicol R. Howard, and Carsten Schulte. Bloomsbury Publishing, 2023, pp. 5–18. DOI: 10.5040/9781350296947.ch-002.

[153] Matti Tedre and John Pajunen. “Grand Theories or Design Guidelines? Perspectives on the Role of Theory in Computing Education Research.” In: *ACM Trans. Comput. Educ.* 23.1 (2022). DOI: 10.1145/3487049.

[154] Chao Tong et al. “Storytelling and Visualization: An Extended Survey.” In: *Information* 9.3 (2018), p. 65. DOI: 10.3390/info9030065.

[155] Jennifer Tsan, Kristy Elizabeth Boyer, and Collin F. Lynch. “How Early Does the CS Gender Gap Emerge? A Study of Collaborative Problem Solving in 5th Grade Computer Science.” In: *Proceedings of the 47th ACM Technical Symposium

on Computing Science Education. SIGCSE '16. Memphis, Tennessee, USA: Association for Computing Machinery, 2016, 388–393. DOI: 10.1145/2839509.2844605.

[156] Sherry Turkle. *Reclaiming conversation: The power of talk in a digital age*. Penguin, 2016. ISBN: 978-1594205552.

[157] Bishakha Upadhyaya, Monica M. McGill, and Adrienne Decker. "A Longitudinal Analysis of K-12 Computing Education Research in the United States: Implications and Recommendations for Change." In: *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. SIGCSE '20. Portland, OR, USA: Association for Computing Machinery, 2020, 605–611. DOI: 10.1145/3328778.3366809.

[158] J. Ángel Velázquez-Iturbide. "An Analysis of the Formal Properties of Bloom's Taxonomy and Its Implications for Computing Education." In: *21st Koli Calling International Conference on Computing Education Research*. Koli Calling '21. Joensuu, Finland: Association for Computing Machinery, 2021. DOI: 10.1145/3488042.3488069.

[159] Eric Von Hippel. *Democratizing innovation*. the MIT Press, 2006. DOI: 10.7551/mitpress/2333.001.0001.

[160] Francila Weidt and Rodrigo Silva. "Systematic Literature Review in Computer Science. A practical guide." In: *Relatórios Técnicos Do DCC/UFJF* 1.8 (2016). DOI: 10.13140/RG.2.2.35453.87524.

[161] David Weintrop, Elham Beheshti, Michael Horn, Kai Orton, Kemi Jona, Laura Trouille, and Uri Wilensky. "Defining computational thinking for mathematics and science classrooms." In: *Journal of science education and technology* 25.1 (2016), pp. 127–147. DOI: 10.1007/s10956-015-9581-5.

[162] David Weintrop and Uri Wilensky. "To block or not to block, that is the question: students' perceptions of blocks-based programming." In: *Proceedings of the 14th international conference on interaction design and children*. 2015, pp. 199–208. DOI: 10.1145/2771839.2771860.

[163] Jing Wu and Der-Thanq Victor Chen. "A systematic review of educational digital storytelling." In: *Computers & Education* vol. 147 (2019), p. 103786. DOI: 10.1016/j.compedu.2019.103786.

[164] Zenodo. *Zenodo*. https://zenodo.org/. Last accessed 2022.

[165] James Zhang, Casey Wong, Nasser Giacaman, and Andrew Luxton-Reilly. "Automated Classification of Computing Education Questions Using Bloom's Taxonomy." In: *Australasian Computing Education Conference*. ACE '21. Virtual, SA, Australia: Association for Computing Machinery, 2021, 58–65. DOI: 10.1145/3441636.3442305.

[166] bell hooks. *Teaching community: A pedagogy of hope*. Routledge, 2013. ISBN: 9780415968188.

[167] bell hooks. *Teaching to transgress*. Routledge, 2014. ISBN: 9780415908085.

Dottorato
di Ricerca in
Informatica

Agnese Addone – Outreach in Computing Education: a Design Toolkit

XXXV Ciclo

AA
2021/2022