

Article

Applying Design Thinking to Enhance Programming Education in Vocational and Compulsory Secondary Schools

Belkis Díaz-Lauzurica ^{*,†}  and David Moreno-Salinas [†] 

Department of Computer Science and Automatic Control, Universidad Nacional de Educación a Distancia (UNED), 28040 Madrid, Spain; dmoreno@dia.uned.es

* Correspondence: bdiaz216@alumno.uned.es

† These authors contributed equally to this work.

Abstract: A proper and complete formation in technology (science, communications, programming, robotics, Computational Thinking, etc.) must be imparted at all educational levels for a lifelong education. However, students may lose motivation or interest due to the complexity and abstraction of some of the concepts imparted. In line with this, the work at hand looks to improve the interest and commitment of students by presenting the programming concepts and contents in a practical way. The teaching–learning process is based on the development of projects about robotics, which are adapted for courses and groups of different educational levels. The Design Thinking methodology is used to impart the content. This methodology allows the students to experiment, design and test different solutions for a given problem, increasing their motivation and interest, promoting creativity, and making the students conscious of their learning process. Two different projects are considered, a simulated one based on a sensor network to localise and track a robot in a closed area for vocational education students, and an experimental one about constructing a robot with several capabilities using Lego Mindstorms for compulsory secondary education students. The results obtained over three different groups of students are analysed and compared, and show that the methodology and projects selected can be adopted and adapted for different educational levels, increasing the proficiency of the students, their development, motivation and self-learning despite the difficulty and complexity of some concepts related to computer science.

Keywords: programming; Design Thinking; compulsory education; vocational education; motivation



Citation: Díaz-Lauzurica, B.; Moreno-Salinas, D. Applying Design Thinking to Enhance Programming Education in Vocational and Compulsory Secondary Schools. *Appl. Sci.* **2023**, *13*, 12792. <https://doi.org/10.3390/app132312792>

Academic Editors: Jenny Pange and Zoi Nikiforidou

Received: 17 October 2023

Revised: 14 November 2023

Accepted: 22 November 2023

Published: 29 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Education in Information and Communication Technologies (ICT) and related fields during the first education levels is essential for training and preparing individuals for the challenges and risks that may rise up in the workplace, social relations, and sustainability [1]. Moreover, a proper and complete education in different science and technology areas must be imparted at all educational levels to prepare the individuals for their incorporation into society and the labour market [2]. This must be accompanied by new teaching strategies in science, technology, and engineering education that enable students to learn new ways of thinking, such as Computational Thinking (CT), which allows them to acquire basic skills in maths and technology and to face problems with reliability [3]. Following this trend, the Computer Science Teachers Association (CSTA) brought up to date the Standards in Computer Science for compulsory education based on the concepts and abilities of CT. However, students may lose motivation or interest, especially those in vocational education, due to the difficulty and complexity of some of the content and concepts. Therefore, it is critical to adapt and modify the teaching–learning process so as to improve the motivation, interest and, finally, the learning outcomes, together with a reduction of absenteeism [4].

The growing interest in competences, learning outcomes, skills and their application in practical contexts shows the importance that the students take an active role and construct

their own learning process, where the teacher has more of a companion role. This approach takes special interest in science and technology subjects. In these areas, the application of concepts in practical tasks is key for a correct formation, and methodologies that rely on the student having the main role in the learning process are becoming of increasing relevance.

These methodologies fit well for students with self-regulating profiles, since they will take care and be responsible for their learning process. Their main characteristics can be summarised as follows:

- Good background knowledge.
- Ability to use cognitive strategies.
- Good time planning and control.
- Avoid external and internal distractions.
- Collaborate in academic tasks and maintain a good classroom climate.
- Motivation to keep attention and effort for studies.

However, these methodologies that require an active role of the students do not fit well in educational levels where students may lack some of the important above characteristics. These students with a more disruptive profile usually show one or more of the following characteristics:

- Lack of important background knowledge.
- Lack of cognitive strategies.
- Inability to plan and control time.
- Inexperience in autonomous working and learning in previous studies.
- Lack of concentration and reluctance to be in classroom, high level of absenteeism.
- Indifference to academic tasks and classroom climate, which may provoke conflicts.
- Lack of motivation since they cannot link the content studied to real life, as well as a lack of study habits with low family involvement.

This latter profile is usually found in basic levels of vocational education (VE), especially in those studies related to technology and computer science. Moreover, in most cases the students who selected these kinds of studies were pushed and/or influenced by family and the media, in which the ICT sector is advertised as a field where it is easy to find a job. The latter, together with the fact that many of VE students did not have success in compulsory secondary education (CSE), negatively affects motivation and makes it difficult to build a classroom climate conducive to learning [5,6].

As in this case it is harder and more complex to apply innovative methodologies based on the construction of learning by the students themselves, the choice is often a traditional teaching model. Usually, it is also the model that these students have already experienced at previous (unsuccessful) education levels, impacting on their motivation. This has a direct negative influence on the classroom climate, which suffers from the conflicts and problems of an unmotivated and uninterested group.

In addition to this motivational and behavioral problem, the complexity of some subjects and concepts from the science and technology field may aggravate this negative situation. The study of any science subject does not only depend on the memorisation of concepts; it is essential to understand them in such a way as to be able to correctly apply the concepts learnt to practical exercises. If the motivation of the students is not adequate, or if even their attitude is disruptive, the complexity of the subject can be viewed as a barrier impossible to overcome.

In an attempt to solve or at least mitigate these problems, this work is based on the Design Thinking (DT) working methodology [7], which is deeply related to PBL. It focuses on analysing an idea or problem in order to implement changes or improvements, or even generate and construct a solution, applying innovation and creativity. It is a process of innovation and, as such, a guideline of applied creativity. Therefore, its main characteristic is to encourage new ways of thinking while dealing with problems. It has been analysed and implemented in recent years in different educational levels, showing that DT can help students to become aware of their creative learning process, motivate

them, facilitate cooperation and teamwork, and boost their creative skills [8]. Although it will be detailed in the following sections, DT, similarly to PBL, enables real and meaningful learning through visualisation, prototyping and testing, allowing the student to manipulate and experiment with real or simulated components, increasing their commitment and motivation. The students can obtain an academic experience that makes them feel that they advance and obtain the skills by their own effort. In addition, it tries to give the students the opportunity to decide over the learning process, selecting and organising part of the work, under the supervision of the teacher. This provides a base for the development of their motivation that reduces the absenteeism.

The DT methodology is proposed for the teaching of computer science at different educational levels so as to increase the students' commitment and motivation, and avoid dropping out of school, especially in the first year of vocational education. This methodology involves practically oriented learning and has shown positive results in previous approaches [9–11]. In particular, DT is used to teach programming to three groups of students belonging to CSE and VE. The content to teach is mainly based on HTML, CSS, and JavaScript, which are explained and imparted in the context of wider robotics projects to increase interest [12]. In this sense, two different projects are considered: a simulated one based on a sensor network to localise and track a robot in a closed area for vocational education students, and an experimental one consisting of constructing a robot with several capabilities using Lego Mindstorms for compulsory secondary education students. These projects will serve as a base for more advanced projects in the subsequent courses.

Therefore, the scope of the present work is to show the applicability and give an insight to the DT methodology to teach computer science subjects in vocational and compulsory secondary education. Based on this background the following research questions are formulated:

Can the Design Thinking methodology be useful to impart computer science contents while increasing motivation and interest, and reducing the absenteeism, especially in vocational education?

Can the students in their first year of vocational education show similar interest and motivation to other groups with better background, commitment, and/or classroom climate if the content is imparted in an alternative and attractive manner?

Then, the contributions of the paper at hand are threefold: (i) to develop a methodology for teaching computer science through the design and development of robotics projects in compulsory secondary education and vocational education; (ii) to show the application of the methodology and its results for three groups of students of different educational levels, analysing the advantages and drawbacks of the methodology proposed in these groups; (iii) to make the students aware of their own learning and to engage them with the use of methods, tools and tasks related to technology and robotics, in order to increase their motivation, commitment and academic development, especially in vocational education. As far as the authors know, there are no studies about the comparison of the teaching of the same contents for several groups of students in different educational levels using a similar methodology and tools.

The rest of the paper is organised as follows. In Section 2 a literature review of the common and new approaches adopted for the teaching–learning process is presented. In Section 3, the groups and context are presented, the methodology used is explained in detail, showing the main features that make it appropriate for its use, and the steps followed and tools used in the teaching process are described. The results obtained with each group are explained in Section 4 and discussed in Section 5. Finally, the conclusions are detailed in Section 6.

2. Literature Review

Recent advances in ICT, robotics, sensors, artificial intelligence, and related fields, have brought new paradigms in work and social relations in an accelerated way. Society in general, and individuals in particular, must face the problems, risks, advantages and

opportunities that these technologies offer and must be prepared for the challenges that will rise up in every aspect of daily life. In order to achieve this goal, lifelong education is a key tool, so a proper and complete formation in technology (science, communications, programming, robotics, CT, etc.) must be imparted at all educational levels.

Regarding CT, it was defined as the thinking process that considers the fundamental concepts of Computer Science to solve problems, design systems, and understand human behaviour [13]. Lately, new nuances have been added to the concept of CT by different authors [14–19], although there is no close definition about the cognitive abilities related to the CT training process. Its main characteristics may be defined as [20]: abstraction; algorithmic reasoning independent of the technology used; division of a complex problem into simple problems; the abstraction automation; debugging process; and the generalisation of solutions. CT has broad application to all disciplines and education levels, such as primary, secondary and vocational education [21–23].

The teaching of ICT concepts such as programming, technology or CT in compulsory and vocational education entails multiple issues that hinder the learning process of students due to the complex and abstract concepts addressed [24]. Moreover, this can be approached in different ways, and currently many countries are developing new education programmes to tackle these concepts [25]. These programmes have in common the objectives of developing the ability to solve problems in daily life, and providing formation related to ICT [20].

In this sense, in the last few years, it is possible to find a significant number of methodologies and educational tools for teaching STEM (Science, Technology, Engineering, Maths) concepts while encouraging motivation, interest, and then the improvement of academic results using ICT [4]. For example, there exist approaches that use educational software or visual block-based programming environments such as App Inventor or Scratch to engage the students [26–29], a combination of software and hardware as in [30] where App Inventor and Lego Mindstorm are used for teaching programming, and robotics [31]. These studies show that when programming is presented in an attractive way, students' motivation and interest increase. As per the methodologies, the range applied is quite broad, but project-based learning (PBL) approach stands out [32], where the students investigate and solve real problems. PBL has shown to be superior when it comes to long-term retention, skill development and the satisfaction of students and teachers, [33]. Multiple authors have successfully tested this methodology, since it makes the knowledge construction process explicit, allows students to be aware of their learning process [34], and is well suited to engineering and technology education [35], although it has been applied successfully to multiple areas, from maths or health sciences to visual arts [36–38]. The students should also have an autonomy that must be adapted by the teacher by designing the project or making a guide. Although it may be complex to deploy, this methodology fits well with vocational education profiles [39,40], increases the motivation, improves the classroom climate and has a direct impact on the personal development of students. This factor is of special importance since, as stated in [41], the basic levels of vocational education have a high ratio of students with disruptive profiles and previous academic failures, showing low interest in academic activities. The reader is referred to [42,43] for a review of PBL works.

Together with PBL, flipped classroom is often applied to increase the interest and motivation of students. There exist some studies where flipped classroom is applied to compulsory and vocational education [44,45]. For example, in [44] flipped classroom is applied to blended learning higher vocational education courses, which require autonomy by the students, showing an improvement in the student engagement. In [45], the authors state that this methodology improves motivation in students of vocational education, but they also indicate that, despite the positive results, the interest and motivation of the students may decrease as the course progresses.

Both methodologies, PBL and flipped classroom, may be difficult to apply in situations where the students present some learning difficulties, disruptive profiles and/or the

climate classroom is not positive. In fact, this could be the reason why there is a lack of research studies that apply flipped classroom and PBL to basic and first levels of vocational education. The present study proposes an alternative methodology based on PBL, Design Thinking, which can be indistinctly used to teach computer science concepts to vocational and secondary compulsory education students.

Apart from the methodology and tools used, it is important to design activities and exercises that are interesting and motivating for the students, as well as to present the content in an attractive way. The goal is that the students look for their own learning instead of just passing their exams, by boosting the student curiosity with questions, problems, and situations that show the usefulness of the content, relating the past content with the new. This also has an important correlation between learning styles and motivation, an essential factor in vocational education and training [46]. The design of these activities can be of a wide variety, such as promoting autonomous and blended learning [47] and including elements of gamification [48]. In addition, the use of new technologies offers new opportunities, such as the use of educational games [49] or virtual reality to increase students' motivation and Computational Thinking skills [50–52].

Moreover, in a group of students, some have different learning pace, due to their motivation, academic reasons or social aspects. In this situation, the teacher has to adjust the materials and activities for the different learning profiles to assure a good classroom climate, which is given by the student–student and student–teacher relations, educational contents, rules, social environment, and even the characteristics of the centre and families. A good classroom climate assures better student–teacher interactions, better behaviour, encourages the learning process, prevents conflicts and avoids situations where some students drop out of their studies. The relation between family involvement and student motivation is crucial to improve the positive climate in the classroom [53]. Therefore, it is essential to develop methodologies, content, activities, and relationships that allow the students to acquire the required knowledge and skills in order to continue their lifelong learning, regardless of their profiles.

Previous work by the authors [25] showed that a methodology oriented to practical and experimental applications, combining robotics and programming, can be very useful and effective to increase motivation and engage students with disruptive profiles and high degree of apathy.

3. Materials and Methods

The protocol adopted for the qualitative research design follows the study [54], and can be viewed as a combination of the three case study approaches presented:

- The description of the case studies is in line with the proposal to describe the particular situations, the context, the instruments used, the data collection and analysis, and the validation.
- The description of the characteristics of the groups, manifesting the objective reality of the participants.

The context and groups are presented below. Then, the methodology and its implementation are explained. Finally, the data collection, analysis and validation are described.

3.1. Context and Groups

The methodology was implemented over three heterogeneous groups of different and well-differentiated education levels. As these are case studies with a limited number of samples, what is proposed in [55] is considered, in which it is explained that “inference about the logical relationship between two characteristics is not based on the representativeness of the sample and therefore on how typical it is, but rather on the plausibility or on the degree of logic of the link between the two characteristics”. In this case, the replication of the results applied to a larger number of participants will be discussed.

The subjects taught to the groups are deeply related, mainly based on HTML, CSS and JavaScript programming. Two of the groups correspond to vocational education (VE) and

the third one to compulsory secondary education (CSE). The methodology was first applied to a group of VE and the group of CSE, and then applied to the remaining VE group so as to validate that results are coherent and consistent. The duration of the units imparted was between 1–2 months. Therefore, the effects of the methodology in the learning process of the students can be evaluated in groups of different features and with different degrees of motivation and commitment. The groups are:

- Group 1: First year of vocational education on Web Application Design (WAD), Brand Language and Business Management Systems subject. In this group, 18 students started the course, 17 applied for the final exam, but only 15 participated in the methodology, since they were not required to attend classes. The age of the students was in the range of 18–20 years, and the group was mainly composed by male students with only two female students. They belonged to families from different social backgrounds: some parents were unemployed, others were university graduates, and some worked in different trades. No students were found to belong to problematic families. They were motivated and the attitude towards effort and work was consistent. Three of the students who participated in the project activity had significant difficulties in abstract thinking and this was maintained throughout time. However, they did not drop out and remained until the end, without obtaining satisfactory results in this respect.
- Group 2: Final year of senior high school of compulsory secondary education (CSE2), Computer Science subject. This group was composed of 13 students and the classroom climate and the commitment of all students was, in general, very good, showing interest in the activities and willingness to work, making this group more suitable for the application of this kind of methodology, as indicated in Section 1, and becoming a good reference to compare with the VE groups. They belonged to upper-middle class families, and most of the students had parents with university grades. All students had ages between 17 and 18 years old.
- Group 3: First year of vocational education on Microcomputer Systems and Networks (MSN), Web Applications subject. The group was composed of 18 students, but 3 of them dropped out at the beginning of course. Five had significant difficulties in abstract thinking, and one was working from home due to a medical treatment. Their social background was lower. Some students belonged to more vulnerable families, with lower income and/or single parents. Others had difficult family situations and suffered from depression. All were male, mostly between 16 and 21 years old, with the exception of two of them who were 25 and 29 years old respectively. Their initial level in programming was quite low, but they kept their interest in the subject.

Therefore, in each group of VE there were students who were eager to learn and performed well; others had difficulties in learning some content (especially JavaScript) and another portion of the students showed no signs of improvement, but were only concerned about passing. Attendance and discipline were stable during the course in the three groups. Therefore, the methodological decisions made were similar in all groups.

The outcomes obtained with these three groups can be partially compared with an additional group in which a classical methodology, such as the master class, was applied. This group, from the first year of vocational education on Administration of Networked Computer Systems (ANCS) was composed of 20 students with very low interest and motivation. The content of the subject was similar to those of the groups studied, especially the HTML unit, which was the same as the one taught to the present groups. Thus, the partial results on this HTML unit can be compared between this ANCS group, where masterclass was applied, and the groups presented above, where DT was used to impart the content. For this particular group, ANCS, the passing rate on HTML was of 55% (only 11 out of 20 passed the exam).

3.2. Design Thinking

In recent years, the use of technology has meant a development of new training processes based on learning instead of teaching. In this way, the student becomes the main actor, being aware of the process, and focusing on action and experimentation.

With this mindset, the methodology adopted is based on Design Thinking, which can be viewed as a creative problem-solving process [56]. This process is composed of a set of phases that are iterated, forwards and backwards, with the objective of acquiring the knowledge required by solving a given problem combining empathy, creativity, and rationality. There does not exist a fixed set of phases, although the most common are: (1) Empathy: to understand obvious or non-obvious needs of the users, the environment and the project. (2) Interpretation/Definition: to analyse the information gathered in the first stage and make a clear and concise approach to the problem, enabling solutions to be found. (3) Ideation: to generate and refine ideas. This stage is the transition between problem identification and solution creation. All ideas are considered valid at this stage. (4) Experimentation: to build prototypes. A solution is designed through trial and error, and with progressive approximations to a satisfactory solution, showing the elements to be erased, improved, changed, etc., until the final solution is reached. This stage is more about experimentation than validation. The aim is to make the ideas developed tangible. (5) Evolution: to evaluate prototypes with the users, to learn from the solution itself and identify possible improvements, failures, what is lacking, etc. It also allows the reworking of solutions or thinking of new ones.

The use of the Design Thinking methodology is closely linked to PBL and can be seen as an extension of experimental and collaborative learning. The students must think about how to solve a problem with the concepts learnt, studying the problem, posing a hypothesis, carrying out an experiment and ending with the analysis of the solution proposed. This iterative process facilitates authentic and meaningful learning through visualisation, prototyping and testing with real problems and solutions. Prototyping is essential in technology education, and it provides new opportunities for the teaching-learning process [57].

Therefore, the aim of this methodology is to measure the degree of assimilation and understanding of the concepts so that the teacher can get to know their students, their needs and difficulties better, generating feedback that allows them to clarify and improve their knowledge [58]. At the same time, it favours interaction skills that will be useful and valuable in future working and/or academic environments. There are several works that have tested this methodology in education and have shown how teachers should apply it [59,60]. However, as far as the authors know, none of these studies have compared the application of this methodology to teach the same content to different groups of students at different educational levels. For a deeper review on Design Thinking on different educational settings, levels and areas, the reader is referred to [61].

In the case of vocational education, or even in compulsory secondary education, students can benefit from a more practical approach, especially those who reject classical methodologies based on the master class, improving their understanding and knowledge on computer science and showing them how the content learned can be used in a real scenario.

3.3. Methodology Implementation

In general, the application of the methodology begins with the presentation of the problem to be solved or the project to be carried out. Its main characteristics, needs and functionalities are highlighted so that students are clear from the outset about the minimum requirements that the proposed solution must meet. No restrictions are imposed in terms of design, tools or ways to reach the solution, so that each student can develop and experiment with their own ideas. In order to complete the project, during the course, several subprojects are proposed to practice the content imparted, which serve as a basis for progressing in the final project. For the content that presents greater difficulties, additional activities should be prepared and greater care and control should be taken in the students' work to ensure that

they acquire the necessary knowledge. In addition, the students should present and defend to the rest of the class their solutions, in order to work on and develop communication skills. Moreover, this allow the students to know and evaluate the solutions developed by their partners, providing new ideas to improve their own work.

In particular, the steps followed in the teaching process were similar to those explained in the previous work of the authors [25]:

- The theoretical part was presented to them, expanding their knowledge through specific questions to the problems posed and that they had to solve by means of individual research.
- In most cases, the sessions lasted 2 h (except in CSE2, where each session lasted 1 h), so most of the activities proposed in the classroom were solved. This allowed the performance of each student to be evaluated.
- A specific example guided by the teacher was explained to show the contents to be learned. Afterwards, they were given a simple example to reproduce with the teacher's help to correct any doubt/error arisen.
- Several gradually more complex exercises related to the final project were then proposed to the students until most of them achieved demonstrable autonomy.
- At the end of the different blocks of the same topic, they worked on a subproject, in which they had to combine the different and new concepts learned, by designing a solution and developing it.
- Where applicable, they were tested on their knowledge by means of a written test.

The choices made and instruments used for the learning process of the three groups were:

- Using daily observation as an instrument to measure the progress of students' work in class and at home. Surveys were also designed to evaluate the students' knowledge at the beginning of the course. Observation, as a data-collection instrument, constituted a scientific technique because [62]: (1) it had specific objectives, (2) it was systematically planned and controlled in relation to general purposes, and (3) it was subject to validity and reliability checks and controls.
- Initially, open-ended interviews were conducted to ask broad questions, followed by semi-structured interviews.
- Use of W3Schools (<https://W3Schools.com>, accessed on 1 September 2023) as the basic website, which stimulated their individual learning at their own pace. The students could consult different web technologies such as HTML, CSS, JavaScript, XML, etc., and also run sample codes in a test environment provided by this site to see how the instructions work.
- Use of Blockly Games to get started in programming and understand basic programming structures.
- Explain the more theoretical concepts in class and review them using Kahoot, a tool also used in [25] with good results and acceptance by the students. The practical rather than the theoretical part of the course was emphasised.
- Contrasting different solutions in class to see the merits or drawbacks of each solution, allowing them to analyse each solution openly.
- Focusing the work on projects, mostly individual, with which to consolidate the knowledge taught. For each exercise/project a preparation/planning time was allocated, which accustomed the students to think about the scheme to be developed and conceive it in advance. Prototypes were made either manually or using online prototyping tools.
- Use Microsoft Teams as a way of communicating with learners to: (a) report on certain events; (b) explain doubts/complaints if necessary; (c) submit work and give feedback on each assignment.
- Informing in advance about the assessment criteria for each assignment, either by means of rubrics or explicitly stated in the assignment instructions. This also ensured the quality of the work presented, by being able to check their solutions before submitting them.

3.4. Data Collection, Analysis and Validation

The analysis of the data gathered was based on the qualitative analysis. Daily activities and their results were recorded in a class diary by teacher and students, combined with observation as a data collection instrument. As previously indicated, this observation was systematically planned and controlled to ensure its validity and reliability in relation to the general objectives and the contents taught [62]. The data collected were carefully analysed to improve the quality of the educative research and to provide more coherence and rigour to the analytical procedures [63,64].

The aspects analysed and evaluated using qualitative methodology were:

- Design conception: the students' activity during each programming session was analysed to assess whether each step was performed correctly. The different solutions proposed were also considered for evaluation, as well as their appropriateness and effectiveness.
- Behaviour during the project: the behaviour of the students was a critical factor in the methodology, as it influences their motivation and class climate, as well as the results.
- Evaluation: results of the examination and presentation of the project, which showed the good performance of the groups.

Data were analysed as they were collected, as suggested by two of the approaches presented in [54], reorganising the data by categories and/or skills to be developed in order to reach conclusions about educational intervention needs. The rigour of this analysis and the triangulation methods is secured by different factors taken into account [65–67]:

1. Dependency: Data were carefully revised and coherent interpretations were reached, paying special care and attention to the data collection process. Direct observation was carried out in a condensed and detailed way, justifying each action and its causes, with easily understandable language. For each activity, the data recorded were: activity, date, time, competences worked on, resources, student development, comments, behaviour, decisions and modifications. Data matching was demonstrated by: (i) the skills acquired and the skills demonstrated during the project; and (ii) contrasting the test results with the analysis of evidence of the skills developed.
2. Credibility: To ensure a complete meaning of the participants' experiences. This fact was especially reinforced when applying the same experience to students of different characteristics and different levels, evaluating the competences based on projects and taking into account the specific characteristics of the students.
3. Confirmation: To ensure the minimisation of the researcher bias. During the different tasks, the same solving method was applied to different exercises and different solving methods were applied to the same exercise.
4. Replicability: To apply and transfer the results (or part of them) to other contexts, avoiding inconsistencies in the conclusions. This point was implemented throughout the experience as the same programming contents (loops, variables, functions, conditionals, etc.) were taught to the three groups under study, keeping the same methodology to promote the motivation.

4. Results

As mentioned before, the methodology was first applied over two groups of different levels for learning HTML, CSS, and JavaScript programming concepts while enhancing their motivation and Computational Thinking skills, and then applied to a third group. Different activities were proposed as part of a bigger project, in which the students had to design, test and evaluate their own solutions and ideas in an iterative process. The goal was to engage the students through a process that involves several subjects and tasks, connecting different contents in a project that motivates them. This project used robotics as context to impart the concepts of programming and to awake the interest of students. The different tasks were adapted to the different profiles and skills of the students, so that, although the content taught is basically the same, the activities may vary from one group to

another in the search of the best motivation and results. The results for the different groups and their corresponding activities are now summarised.

4.1. First Year of Web Application Design in Vocational Education

The project proposed to the 15 students of the first year of vocational education in Web Application Design (WAD) was in the context of the Brand Language and Business Management Systems subject. The project consisted of simulating the localisation and control of several robots so that the students could apply the concepts learned about HTML, CSS, JavaScript, XSD, DTD, XSLT, and XML. The assimilation of these contents and concepts, as well as others from other subjects (math operations, variables, loops, functions, etc.) is obtained by designing, testing, evaluating and validating their own solutions, and comparing and discussing them with the solutions of their classroom mates.

The statement of the project was based on a robotics shop, where several robots are exhibited and each of them has an area to demonstrate its skills. As only one of them can be sold, their skills should be tested and evaluated to select the most appropriate. Therefore, the students should define an application with the following characteristics:

- Delimit the area in which each robot could act and move, and obtain its position, as part of an indoor localisation system.
- Test the connection of JavaScript and XML to read data from a file. Store and save the robot movement in a XML file, duly validated by DTD or XSD.
- Control that each robot only moves in its corresponding area and the visualisation of data.

Apart from the above, the results of the scores given to the robots should be presented in the correct format using XSLT.

Notice that additionally to the HTML, CSS, and JavaScript concepts, the students had to apply additional knowledge and skills acquired in other subjects and related to programming and CT. In this sense, the robot position is computed based on a real positioning system, where the distances from a set of sensors to the robot are measured, and based on these distances, the robot position is determined. In the simulation project, the sensors are considered to be placed on the four corners of the area in which the robot can move, so students should apply trigonometry and maths operations to compute the position. In addition, this position must be computed recursively to update the robot position based on the programmed or commanded movements, so loops, variables, functions, etc. must be correctly applied to obtain a functional and valid solution.

Before working on the project, the students were working on different activities to acquire enough skills and demonstrate their proficiency in HTML, CSS, JavaScript, the programming and validation of XML files with DTD and/or XSD, and their corresponding presentation with XSLT. Partial developments of the robot project were also carried out in each unit so they could appreciate the possible application of these concepts to the project, although they were not the final designs or decisions adopted. However, these partial developments allowed them to start with the testing and evaluation of prototypes and ideas. Some of these tasks are presented next.

HTML: Students had no prior knowledge of HTML at the beginning of the course. Therefore, their first steps focused on the creation of links (to web pages, or a section inside a document), insertion of images and video considering the specific MIME types and using relative and absolute addresses, tables and forms for data entry, accessibility and semantic tags on pages, etc. It was important to start with HTML, firstly for its accessibility and ease of use, and secondly in order to gain abilities regarding design and conception for the final project.

JavaScript and CSS: Initially, a set of exercises directly related with the project were proposed to the students, which allowed them to work with the interface, to position objects on the screen and also to react to different events. Different activities were included both for handling the page and to operate with data types such as vectors or strings using control structures. Once these concepts were assimilated, CSS tasks were carried out to

style a HTML document, together with JavaScript functions to provide more dynamism to the page front-end and a responsive design.

For a better understanding of these concepts, the students had to develop from scratch a “cultural magazine” web page using HTML, CSS, and JavaScript. The resultant HTML document should keep an easily maintainable structure with good search engine ranking capabilities. Mandatory elements were including links of different types, a form, some of the lists (ordered, unordered, definitions), a table, tags that make the page more accessible and understandable for browsers, and images and videos. In Figure 1, one of the solutions developed by the students about the cultural magazine is shown. Most of them achieved the requested requirements.



Figure 1. Example of the cultural magazine developed by the students. This example fulfilled all requirements asked.

Learning HTML was well received by the students, but CSS and JavaScript were somewhat more difficult for some. Nevertheless, the pass rate in the individual exams that were taken at the end of each content was over 50% in both, as detailed in Section 4.3. The student difficulty perception for each subject can be seen in Table 1.

Table 1. WAD students’ perception of content: easy (E), moderately easy (ME), moderately hard (MH), hard (H).

Content	Student Perception
HTML	ME
CSS	MH
JavaScript	MH

Regarding learning difficulties, the main one was the learning and understanding of the programming structures, especially loops. This was a common difficulty in all groups, and more time was needed to understand and adequately handle these concepts. At this point, the JavaScript concepts were not mature enough yet and had to be reviewed later on.

XML files and validation: Concepts related to databases already taught in the databases subject, such as 1:1, 1:N, and M:N relationships, were reviewed, confirming that they had this knowledge. The differences of XML as a tag language with respect to HTML were established and files with different hierarchies were created. Reflecting the characteristics of the users with their nickname, age and password; organising and storing products in a greengrocer’s/gardener’s shop or simulating the organisation of a magazine or class schedules were some of the exercises carried out and reviewed in the classroom. Regarding the project of the robot, the students started to deliver different XML solutions, which allowed the specification of several areas and several robots with their functionalities and spaces where they can operate.

To check that the XML files were defined correctly, DTD (Definition Type Document) and XSD (XML Schema Definition) were used interchangeably. In both cases, emphasis was placed on the relationship that should exist between the definition of these schemas

with the XML, so the students always first made the XML for each validation exercise individually. In this way, the practice in generating XML files was reinforced.

DTD: How to define attributes and elements, how to refer to an internal or external DTD, as well as how to specify keys while maintaining integrity, were the issues basically addressed in the DTDs. In addition to the previously mentioned exercises to create an XML, to which the corresponding DTD was included, the exercise related to the robot shop was carried out. In the solutions requested, the positions, the selected robot and the plot should refer to an existing value, so as to validate possible errors in the data entry in the XML.

XSD: Although most students understood DTD better and saw it more accessible, others opted to validate the XML file with XSD, as they perceived it as a more exact way of defining the elements and attributes, although they also considered the .xsd file to be much more extensive and that could be cumbersome. Some of the exercises carried out previously were taken up again to validate them in this way. Subsequently, they were reinforced with new exercises. They were trained in: (a) defining simple elements with different attributes; (b) the validation of elements composed of other elements; (c) the cases in which it was more convenient to declare attributes; (d) ensuring the referential integrity of the elements, so that no element was referenced that had not been in the system before; (e) comparing different solutions, encouraging discussion about the merits of each one; (f) referencing the .xsd document from within and outside the .xml. The results obtained in the exams showed proficiency in these types of problems and they proposed very complete solutions.

XSLT: to display the elements of a .xml file and for timing issues, the structures “xsl:for-each”, “xsl:if”, “xsl:when”, and “xsl:value-of” were used. In the robot shop project, the users participating in the voting and robot selection should be shown. To determine which robot won the voting, the condition was set for each of the participating robots. In Figure 2 one of the solutions is shown. In Figure 2a the test of robot movements is done and in Figure 2b the voting for the best robot is shown. To make another movement, the XML data are changed and when the button is pressed again, the movement continues from the previous position. The “Access voting” button opens the page with the voting data of the XML made with XSLT. The XML validation DTD was included.



Figure 2. Example of one of the project solutions. (a) Areas for the robot movements and forms to enter the commands; (b) Page for voting the best robot and for showing the results.

The latter concepts, which were only taught to this group, were evaluated through an exam whose results are shown in Table 2.

Table 2. Examination results in WAD students about XML, DTD, XSD, XSLT contents.

	Students	Passed Exam/Retake	Examination Pass Rate in Exam	Total Examination Pass Rate
DTD	15	8/3	53.3	73.3
XSD	15	6/2	40.0	53.3

At this point, and after learning how to format the outputs coming from a .xml file, the students were able to take on completely the research project. The results showed the different opinions of the students when conceiving the robot’s movement, which served as a basis for the localisation algorithms used. These algorithms were based on basic trigonometry using the distances measured to the robot from conveniently spaced sensors. The adequate sensor configuration and target position estimation followed descriptions similar to those in [68,69]. As mentioned before, in the simulated project, the sensors were considered to be placed at the four corners of the area of each robot, and the robot position was iteratively computed by the use of trigonometry, math operations, variables, loops and functions. Mainly, they had difficulties in connecting JavaScript and XML, as well as in conceiving the movement of the robots at the same time. These tests served to draw conclusions about how best to approach the teaching of programming in these areas. Not all students were able to solve the proposed localisation problem.

4.2. Second Year of High School Secondary Compulsory Education

A similar experience in terms of teaching HTML, CSS, and JavaScript was implemented in the second year of high school compulsory secondary education (CSE2), where the learning of HTML was also better accepted than CSS and JavaScript. A project was proposed so that they could apply the knowledge and skills learned during the course. Although they were taught JavaScript to be able to manipulate the pages, the level of commitment of the students at the stage of the course (final weeks) was lower, and taking into account that it was an optional subject and the project was also optional, the concentration decreased. Nevertheless, the results are considered to be also satisfactory, and the generally good classroom climate and good students’ background and commitment facilitated the application of the methodology, with all students participating in the project although it was optional, as mentioned.

Different activities were also proposed to practice the contents before addressing the final project. Similarly to WAD students, they were asked to apply what they had learnt in HTML during the first term by imagining that they were managers of their own companies and should perform marketing functions in order to publicise their company’s potential. To do this, they had to design a web page, through which they could disseminate the main results of the company. As a requirement, they also had to include a form, a table, lists, images and videos, semantic tags and good search engine positioning. Emphasis was placed on the internal structure of the document, so that it would be easily maintainable. The activity was successfully completed by most of the students, and one of the solutions proposed is shown in Figure 3.

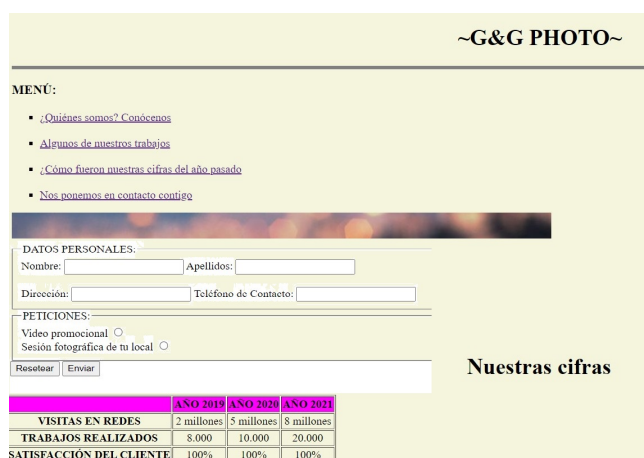


Figure 3. Example of the web page developed by the students. This example fulfilled all requirements asked.

Subsequently, learning of CSS was included, practising first with the previous pages. They had to apply what they had learned in HTML and CSS in terms of formatting and positioning. The pages had to show the activities prepared by each region of the country of the company.

A project was proposed to incorporate and apply all the knowledge acquired. This project was based on the same contents, i.e., HTML, CSS, and JavaScript, applied to robotics and programming. It was presented as end-of-year project and it was composed of three steps:

1. Search of information about computer security.
2. Construction of a robot considering the concepts learned.
3. Web page construction.

The DT methodology was mainly applied to step 2 and 3. In step 2 the students had to construct different robot prototypes, testing different ideas and solutions, evaluating and comparing them, and finally selecting and validating one for the final prototype. In step 3, they had to construct the web pages following a similar procedure to the one indicated in the WAD group.

Then, following the above steps, the students were asked to carry out research in groups on computer security and awareness of active and passive safety through robotics. They had to look for information on applications that generate security breaches (a specific application was assigned to each group) and how to implement protection by applying active and passive security. Three teams were created, where each one had to deliver a document with its research and also present it orally in the class. Then, they made a Kahoot test to reinforce the knowledge imparted and subsequently, they proceeded to implement in 2 groups an interpretation of the safety concepts using Lego Mindstorm. The two final prototypes of the robots constructed by the groups are shown in Figure 4. All was reflected in a web page that each team made, in order to show the impressions of their peers on the process of making the robot, as well as the results obtained after applying the concepts learned in the project.



Figure 4. Robots implemented using Lego Mindstorm. (a) Robot of group 1. (b) Robot of group 2.

It is important to mention that before addressing control structures in programming (loops, conditionals) for the robot movements, they were proposed to work with Blockly Games (<https://blockly.games/>, (accessed on 1 September 2023)) based on positive experiences from previous years [25].

Although they also included programming sentences in their pages, it was considered that their knowledge of programming was very limited for several reasons: they did not understand loops well, it was necessary to dedicate more time to this section than the available in the didactic programming, and the students' dedication at this stage of the course was not complete, as they had to prioritise other compulsory subjects (this one was optional).

The difficulty perception in this group can be found in Table 3.

Table 3. CSE2 students' perception of content: easy (E), moderately easy (ME), moderately hard (MH), hard (H).

Content	Student Perception
HTML	ME
CSS	MH
JavaScript	H

4.3. Evaluation

Rubrics and assessment criteria were used based on the previous years experience. The results of the final assessment take into account the students who did the exams (HTML, CSS, and JavaScript), considering that at certain points in the course some of them dropped out for different reasons in VE (which is why sometimes the number of students evaluated does not coincide). In Tables 4–6, the examination results for the different units in WAD, in CSE2, and the final pass rate in the course for both groups are shown, respectively.

Table 4. Examination results in WAD.

	Students	Passed Exam/Retake	Examination Pass Rate in Exam	Total Examination Pass Rate
HTML	18	16	88.8	88.8
CSS	17	7/3	41.17	58.8
JavaScript	17	7/3	41.17	58.8

Table 5. Examination results in 2nd year of high school CSE.

	Students	Passed Exam/Retake	Examination Pass Rate in Exam	Total Examination Pass Rate
HTML	13	11/2	84.6	100
CSS	13	10/3	76.9	100
JavaScript	13	9/4	69.2	100

Table 6. Examination results in the subject/course for CSE2 and WAD groups.

Subject/Course	Students	Passed	Examination Pass Rate in Exam
CSE2	13	13	100.0
WAD	17 (15)	12	70.6

It can be seen that in HTML, both groups obtained better results than in CSS and JavaScript and that they did indeed find it easier to follow. In the case of CSS, they generally obtained higher marks than in JavaScript, the latter being very tight, close to the minimum. Notice also that for WAD students, the marks of the final exam for CSS and JavaScript are very similar. However, the final scores are consistent with the partial results observed in class, which showed that the concepts and knowledge were correctly acquired.

5. Discussion

The disparity in the characteristics of the students was evident, where some had no skills or motivation to learn; others showed interest and dedication of effort and time to perform such tasks. Therefore the tasks, and even the projects, had to be adapted to involve all profiles and increase the motivation, interest, and commitment of students. This fact was more evident in vocational education, where, as mentioned in previous sections, there are more disruptive profiles, i.e., students that are not interested in the course or subject, and

they are neither motivated nor interested in learning and participating in class. The latter may even affect negatively the class climate, producing a contagion effect on the rest of the classmates. For this latter reason, it is very important to orientate the teaching–learning process using methodologies that involve and make these students feel important, such as the Design Thinking considered in the present work.

This project-based methodology has been demonstrated to be quite effective in increasing motivation and interest in the students as indicated in [8] and confirmed by the results obtained, where the students showed interest, participated actively, and the absenteeism was very low, with with 100% positive marks in compulsory secondary education and almost 65% in vocational education. The content was introduced progressively, giving the students the opportunity to try out partial solutions related to each portion of the content and concepts. The objective at those moments was not to develop partial solutions that would become part of the final project, but to become fluent with the computer science concepts and their programming, try out ideas and test them in order to consolidate the concepts and knowledge, and gain confidence to develop the final project. Moreover, these solutions were discussed in class, so that the students could understand the different approaches taken by their classmates, observe different designs and ideas, and boost their creative process.

The students considered this experience very positive, as they could see the use of each concept in a real project, connecting different contents and lessons, and understanding the importance of each part to reach a global solution, similarly as indicated in [9–11]. However, these references were applied to higher vocational education, and as mentioned in Section 1, there is a lack of works applying this kind of methodology in the initial courses of VE, possibly due to the students' characteristics of this initial years of VE. In this case, the application in the first course of VE was very well received by the students, motivating and engaging them. They were very involved in the learning process, asking for more exercises and even working on and developing their own projects (personal web pages). The absenteeism was very low, with 3 out of 18 students that did not participate in the project and did not regularly attend class. Regarding CSE, the group was very disciplined and had already a very good classroom climate, so the application of this methodology was easier. However, the commitment and interest shown by the students were similar in both groups, showing an improvement of interest, motivation and classroom climate in VE, which is a critical factor as stated in [41].

Moreover, the context of the projects, about robotics and programming, was also an important factor to engage students and raise their interest, which also allowed us to exploit the concepts of Computational Thinking in some lessons and activities, so that complex concepts of programming could be better understood and internalised.

Regarding answers to the research question raised:

- Can the Design Thinking methodology be useful to impart computer science content while increasing motivation and interest, and reducing absenteeism, especially in vocational education? As previously indicated, the VE students considered this experience very positive, and they were involved during the teaching-learning process, being very active and participating in class. They designed, tested, and developed their own solutions to the project proposed, and they even continued working on their own on personal projects. The absenteeism was very low, with there being only 3 out of 18 students that did not attend almost any class during the course. Moreover, attendance was not mandatory and they could just attend the final exam, however most of the students participated in the project actively.
- Can the students in their first year of vocational education show similar interest and motivation than other groups with better background, commitment and/or climate classroom if the contents are imparted in an alternative and attractive manner? The students of VE showed similar interest and commitment to CSE students. The classroom climate improved significantly and they showed real interest in the process of working in projects and developing solutions to acquire the knowledge. They considered this

learning process more useful than classical methods, since they felt that they could apply the content to real life. The above showed that with the adequate methodology, the students can gain interest in their studies.

Next some additional recommendations, results, and problems are discussed in deeper detail.

5.1. Methodology: Practical Implementation and Recommendations

Some practical recommendations and guidelines that can serve as guide for other researchers/teachers are now highlighted.

From the first moments, it is necessary to clearly explain the project and its main requisites, so that the students have no doubt about what they need to develop. It is important to mention the different steps to carry out until the final prototype is reached and what content and concepts are needed for each part. The clarification of these points, as well as showing the practical orientation of the lectures, helped to improve the classroom climate and to reduce the absenteeism of the first week. Once the initial weeks passed and the students could check the real nature of the project and its practically oriented approach, the assistance given to the class remained stable.

Approaching each of the topics with a demonstrative exercise, first guided by the teacher, and then proposing exercises with an increasing degree of complexity in which they gained more and more autonomy on a given topic or project proved to be a good method, as the results show. Each new piece of content was also related to the final project, so that the students could design and test possible solutions to specific requirements of the final project.

When explaining a new topic, they were asked to reason why one solution was more viable than another to promote their creativity, showing the advantages and/or difficulties of each solution. It was therefore crucial to keep an eye on the progress of each student in the classroom, in order to detect errors when dealing with the solutions.

Analysing and discussing the students' own solutions on the digital whiteboard was important for those who had difficulties in obtaining an idea of what their classmates were doing. Moreover, it allowed the students to appreciate different solutions, approaches or designs that could be openly discussed, providing the students with new ideas to test and practice.

Contrary to what is stated on p. 49 of [70], the students were encouraged in this part of the course as it was very practical and they even started to make their own personal web pages. This good behaviour and interest in practising the content resulted in a better classroom climate.

Regarding the specific concepts and content imparted, the order in which they were taught was appropriate. Starting by teaching HTML is a good way to attract students' motivation, especially when they start from scratch, as they do not perceive it as something difficult. Introducing CSS and JavaScript afterwards is organic, as well as the posterior teaching of XML, DTD, XmlSchema, XSD, XSL, etc. Perhaps it would be convenient to shorten the period of teaching CSS to approach it mostly through JavaScript. This would improve the understanding of the programming logic. Students intuitively formatted pages, so among the contents those related to the positioning of the elements, which are less comprehensible to them, would be the most important to teach. This would allow the start the study of JavaScript before. Likewise, it would be interesting to give more prominence and time to XMLSchema, as it offers more secure options for defining XML than DTD.

5.2. Common Conclusions to the Two Groups

Some common conclusions can be stated for both groups, since they received the proposal and exercises with similar interest and motivation, and they put similar effort and commitment into their corresponding activities.

5.2.1. HTML, CSS and JavaScript

In general, although WAD students showed better performance in all the topics analysed (HTML, JavaScript and CSS) due to their academic level and vocational training course, all students perceived HTML as more accessible to learn. It is important to note that the complexity level of the contents for WAD students was significantly higher, which is one of the reasons for the lower marks of the group despite their good performance.

The time spent for the initial planning of the proposed pages and projects was very important for the final result. Emphasising the relative and absolute directions of the elements was crucial for the portability of the applications.

In the case of CSS, a clear distinction was made between the students who were more inclined towards design and those who were more inclined towards programming. The time and exercises dedicated to responsive design and the positioning of the elements on the page were sufficient to undertake the final project. They became used to handling .css files instead of implementing them directly on the pages, which made them more flexible and easier to maintain.

JavaScript was taken up as a challenge by the majority of WAD students, while the CSE2 group of students perceived it as very difficult, also due to the short time planned.

5.2.2. Tools: W3Schools, Blockly Games, Kahoot

The use of W3Schools for learning (especially at the beginning) about the different technologies was very well received, as the students appreciated all the resources it offers, including the possibility of trying out the examples on the site itself. It was especially important at the beginning of each topic and during the development of the initial exercises, where they were still dependent on the teacher's advice. At this point, it was a very useful tool for assisting several students at the same time, according to their own pace. With regard to Blockly Games and Kahoot, they provided satisfactory results similar to those shown in [25].

5.3. Problems Encountered

Despite the good results obtained in both groups, during the application of the methodology some problems and difficulties arose.

5.3.1. Common Problems to Both Groups

Lack of abstraction in the solutions and/or hasty solutions without a previous plan in the CSE2 students: In the case of JavaScript programming, they started to program directly without planning or thinking on the solution beforehand. While it is true that in the case of the page design they made a prototype beforehand, on the JavaScript programming they lacked planning, possibly due to the short class period dedicated to this task.

Problems with loops structures: This was a recurring problem in all groups. It was difficult for them to identify when to use them, although WAD students understood these concepts more quickly.

Responsive design test: Although they implemented within the pages the way of visualising them on different platforms, they did not carry out the corresponding tests to verify it.

Cross-browser testing: This requirement was requested on several occasions, but they did not recognise the value of doing so and did not perceive it as crucial.

Difficulties in acquiring knowledge by reading new documents: In the students in later years, deficiencies were detected in the way they expressed themselves orally and in writing. For this reason, the tasks requested were to be delivered together with documentation explaining the process for obtaining the different results.

Incorrect validations: In HTML, they designed forms without a submit button or <form> tag, they did not correctly validate required fields, and they selected incorrect data types.

Requirements not taken into account and problems in language competence: In tasks where the statement was very specific, at a formal level they found it difficult to include the requested requirements, such as concrete tests/tasks names, interpretation of instructions, refusal to read a lot of text and/or additional bibliography, use of a large variety of colours, solutions without taking into account what the supposed client states, etc.

5.3.2. Main Difficulties with XML, DTD, XSD, XSLT in WAD.

These concepts were only imparted to WAD students, so these difficulties only concern this group. The main difficulties encountered in the validation of the .xml files were:

- Identifying the correct validated representation: designing the well-formed .xml was difficult at first, especially in defining what level of granularity to achieve.
- Based on how they defined the .xml, they had to achieve its validation, which took time to understand, since they first had to take up the different relationships (1:1, 1:M, N:M) that they wanted to represent, looking for a way to do it, without making their validation more complex. This prompted several discussions in the classroom, so that they could see the different variants they could have.
- Adhering to the statement and to what was requested: their solutions sometimes did not follow the instructions.
- Lack of validations, cardinality, or referential integrity.

Even so, and taking into consideration how complex the validation of the .xml files was for some students, especially in XSD, the students showed a fairly high degree of commitment and asked for more exercises.

5.4. Application to Other Groups: First Year of Networks and Micro-informatics Systems in Vocational Education

Given the good results obtained in the two previous groups, it was decided to apply the methodology following the same steps and ideas to a different group so as to confirm the good learning outcomes obtained. This new group was from the first year of vocational education in Networks and Micro-informatics Systems (NMS). Similarly to the WAD group, an HTML, CSS, and JavaScript exercise was given to students of NMS and the results were satisfactory. JavaScript was tackled to a lesser extent, given the time planned for this activity and the difficulties shown by the students, whose training was more hardware-oriented. However, when they had to learn computer applications that generated HTML code, as was the case with Wordpress or Joomla, in many cases they preferred to code by themselves using what they had learned in the first part of the course. On this occasion, after learning CSS, they made a Christmas web page, instead of a cultural magazine. The difficulty perception in this group is shown in Table 7, which is equivalent to the perception of the CSE2 group.

Table 7. NMS student’s perception of content: easy (E), moderately easy (ME), moderately hard (MH), hard (H).

Content	Student Perception
HTML	ME
CSS	MH
JavaScript	H

These students were also asked to carry out a research project to position several elements at the same time in a given area, similarly to that of WAD students in Section 4.1. The contents were gradually explained, adding complexity and resources to the solution requested. The results obtained are shown in Table 8.

Table 8. Examination results in NMS.

	Students	Passed Exam/Retake	Examination Pass Rate in Exam	Total Examination Pass Rate
HTML	15	13/1	86.6	93.3
CSS	15	10/3	66.6	86.6
JavaScript	15	11/2	73.3	86.6

These results are compared with those obtained for WAD and CSE2 in Table 9.

Table 9. Examination results in the subject/course for CSE2, NMS and WAD groups.

Subject/Course	Students	Passed	Examination Pass Rate in Final Exam
CSE2	13	13	100.0
NMS	15	13	86.6
WAD	17	12	70.6

It can be seen that the results were very good, with almost 85% of students achieving positive marks and passing the exams. Moreover, they had to learn applications to generate web pages, and whenever possible, they opted to create the code themselves, avoiding as much as possible to generate it through these applications. These results improve significantly those obtained with WAD students, which may be due to refinement of the methodology, knowledge of where problems may arise, and better guidance and monitoring of the project based on previous experience. The results are slightly worse than those obtained in CSE2; however, it must be noted that in CSE2 the content is not taught in such a profound way as in vocational education, and the CSE student profiles show significant differences, with better initial commitment and classroom climate.

The main comments stated in Sections 5.2 and 5.3 are applicable to NMS students, so that the application of the methodology provides stable and coherent results.

It is possible to compare the results obtained in the HTML unit for the three groups with the results obtained for the group ANCS, which studied this content using the master class as main teaching methodology. The ANCS group had a pass rate of 55% in HTML, while WAD, CS2 and NMS had a pass rate of 88.8, 100, and 93.3%, respectively, see Table 10. The improvement in all groups, especially in those of vocational education, is very significant even in a relatively simple unit as HTML programming.

Table 10. Examination results in the HTML unit.

Subject/Course	Students	Passed	Examination Pass Rate in HTML
ANCS	20	11	55.0
WAD	18	16	88.8
CSE2	13	13	100.0
NMS	15	14	93.3

Regarding the final grades, in Figure 5 the distribution of marks obtained by the students of each group is shown. It is possible to notice how a larger number of students passed the course and with better grades in the groups where Design Thinking and robotics projects were used. In addition, the mean, median, standard deviation and mode of the final grades of each group are shown in Table 11. It must be noted that the purpose for providing this information is to give a more detailed description and for comparison purposes of the results among the three groups in which the methodology DT was applied (WAD, CSE2, and NMS). As mentioned above, the comparison between ANCS group and the rest of the

groups is only valid for the HTML unit (shown in Table 10), since for the rest of the course, the content was different. However, the information about the final grades of the ANCS group is included for the sake of completeness.

Table 11. Descriptive statistical analysis of the final exam results.

Subject/Course	Pass Rate	Mean	Median	Standard Deviation	Mode
ANCS	50	4.82	5	2.91	1
WAD	70.6	5.41	5	2.93	5
CSE2	100	8.5	8.9	0.92	9
NMS	86.6	6.33	7	2.06	6

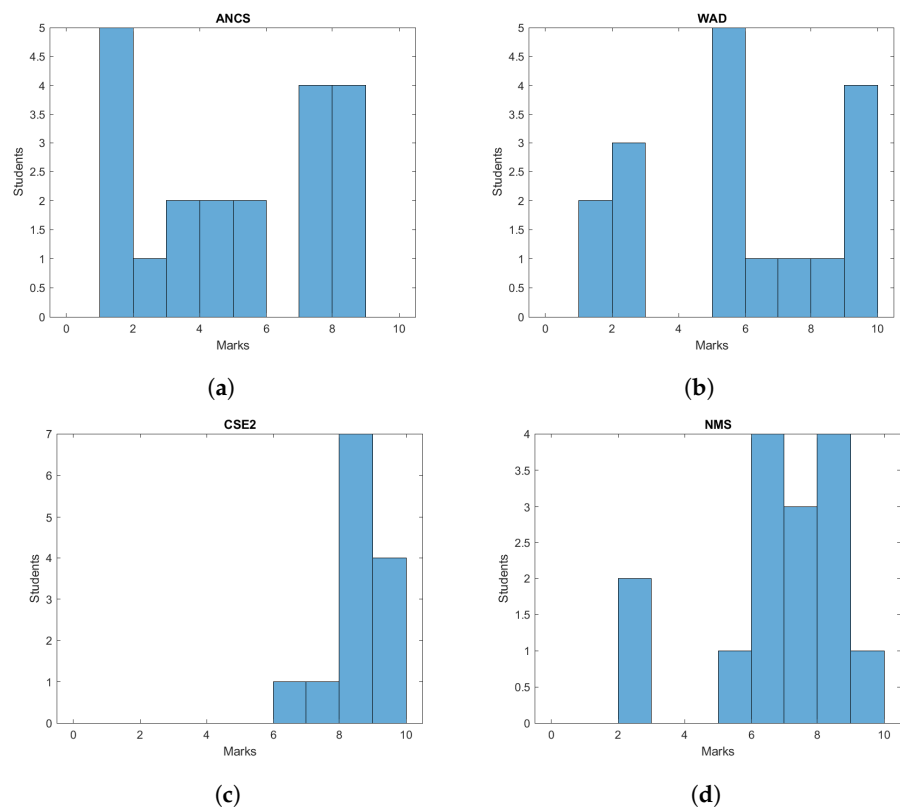


Figure 5. Distribution of the final grades in the groups. (a) Final grades of ANCS group. (b) Final grades of WAD group. (c) Final grades of CSE2 group. (d) Final grades of NMS group.

The improvement on the final grades of the students can serve as indicator of the success of the methodology applied, but it is important to remark that the marks are only an additional factor that confirms this success. The success of the methodology is mainly based on the interest that the students showed, their commitment in all activities and projects, the improvement in the classroom climate, and the interest they showed in developing their own projects. Moreover, the absenteeism was significantly reduced, a very important and critical factor, especially in vocational education. Therefore, it can be stated that the methodology proposed, Design Thinking, as well as the project context, robotics and programming, allowed students to increase their interest and motivation to learn the subject, to better understand the content and concepts taught, to incorporate them into a project that justifies their learning and use, and to become responsible for their own learning process by seeking designs and solutions to the problems posed.

5.5. Limitations

The application of the methodology to different educational levels and subjects can be viewed as a limitation, since some of the results and conclusions may be different due to the heterogeneity of the groups and they cannot be fully compared. However, it allows us to draw general conclusions about the applicability of the methodology for programming learning, and as indicated in Section 3.4, the replicability of results in different context confirms the suitability of it.

In addition, the total number of 45 students is a limitation and general conclusions cannot be stated; for this reason, this work is considered a case study. In this sense, the aim is not a generalisation of the found outcomes, but to give insight into alternative methodologies and proposals that can raise motivation and interest in vocational education students. The results obtained are limited to the groups studied, but the outcomes and conclusions obtained can serve as a guide for teachers that have to deal with similar situations to promote the interest of the students. This goal is based on the intrinsic characteristics of the qualitative research and the incidental sampling of the participants involved in the experience, so that the outcomes and results may be transferred or applied to similar and different contexts. This transference, or applicability of the outcomes, in qualitative research is not a generalisation but the application of the essence of the findings to other contexts, provide some guidelines to obtain a general idea of the problem studied and apply some solutions to other situations [71]. Moreover, the proposal can be adapted and modified depending on the different learning styles and profiles of the students and previous experiences.

6. Conclusions

In this paper, the methodology Design Thinking has been applied for teaching concepts and content in the computer science field, such as HTML, CSS, and JavaScript, and the corresponding programming structures and techniques. To impart these concepts, a global project to relate them was presented, and the students were required to design, test, and develop solutions to this project. The projects were focused on a robotics application, to increase the interest of the students and to create a context rich enough to impart concepts of programming and Computational Thinking in an attractive way. The concepts were sequentially presented and used to add complexity and functionalities to the solutions designed, until the final solution was reached.

The results obtained for two groups of different educational levels, one of high school secondary compulsory education and another from the first year of vocational education, showed that the methodology proposed was well received by the students and it motivated and interested them, showing good results in the classroom climate and exams, and reducing the absenteeism. The students participated actively in class, developing and discussing possible solutions, the absenteeism was very low, and their involvement in the learning process was very significant. The results of the HTML unit could be compared with another group of vocational education with a similar subject in which a classical methodology, such as the master class, was applied. The results in the HTML unit, which was exactly the same to the groups under study, improved significantly, going from 55% positive grades in the control group ANCS, to 88.8% in WAD and 100% in CSE2, together with a better classroom climate and participation of students in the different activities. Moreover, this methodology was subsequently applied to a new group of the first year of vocational education, obtaining even better results in students' behaviour and final marks than in the previous vocational education group, thanks to the experience acquired and to have identified the different problems and difficulties that may arise, allowing a better orientation and preparation of the content to be imparted.

Although the results are obtained over a small group of students, this methodology in which the students are leading their own learning by designing, testing, and developing a solution to a proposed project or problem, allows an increase of motivation and commitment, even in disruptive profiles. Based on the qualitative research and on the results

of other qualitative works [72], the outcomes could be transferred to similar or different contexts. If this transference or applicability to other context would be difficult, the results could be used as guidelines to outline a general idea of the problem and to apply some solutions in other contexts if there are enough similarities to transfer the results [65,73]. In this sense, it has been shown that the methodology can be easily extended to other courses. The possible criticism of the weak generalisation can be overcome by expanding the local knowledge accumulation with a larger number of qualitative studies [74].

For future works, this methodology will be tested and applied for different concepts and content related to computer science, considering also different educational levels to evaluate the applicability of the methodology in the widest sense possible. In particular, the design and development of a hardware and software platform for indoor localisation will be completed, similar to the one described in the simulation examples of this work. With this platform, the students of vocational education could carry out hardware and software projects, as well as learn programming in a bigger project in which they could apply all the content of a subject and even connect and relate different subjects.

Author Contributions: Conceptualisation, B.D.-L. and D.M.-S.; methodology, B.D.-L. and D.M.-S.; software, B.D.-L.; validation, B.D.-L. and D.M.-S.; formal analysis, B.D.-L. and D.M.-S.; investigation, B.D.-L.; resources, B.D.-L.; writing—original draft preparation, B.D.-L. and D.M.-S.; writing—review and editing, B.D.-L. and D.M.-S.; visualisation, B.D.-L. and D.M.-S.; supervision, D.M.-S.; project administration, D.M.-S.; funding acquisition, D.M.-S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the “Ministerio de Ciencia e Innovación” of Spain under project PID2020-112502RB-C44. Part of this work was also supported by Innovation Group “IEData” GID2016-6.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy reasons.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

ANCS	Administration of Networked Computer Systems
CSE	Compulsory Secondary Education
CSE2	Second year of Compulsory Secondary Education
CSS	Cascading Style Sheets
CSTA	Computer Science Teachers Association
CT	Computational Thinking
DT	Design Thinking
DTD	Document Type Definition
NMS	Networks and Micro-informatics Systems
PBL	Project-Based Learning
STEM	Science, Technology, Engineering and Mathematics
VE	Vocational Education
WAD	Web Application Design
XML	Extensible Markup Language
XSD	XML Schema Definition
XSLT	eXtensible Stylesheet Language Transformation

References

1. Staniškis, J.K.; Katiliūtė, E. Complex evaluation of sustainability in engineering education: Case & analysis. *J. Clean. Prod.* **2016**, *120*, 13–20. [[CrossRef](#)]
2. UNESCO. *Using ICTs and Blended Learning in Transforming Technical and Vocational Education and Training*; Commonwealth of Learning: Perspectives on Open and Distance Learning; Latchem, C., Ed.; UNESCO: Paris, France, 2017.
3. McLaughlan, R.G. Instructional strategies to educate for sustainability in technology assessment. *Int. J. Eng. Educ.* **2007**, *23*, 201–208.
4. Khan, S.H.; Markauskaite, L. Approaches to ICT-enhanced teaching in technical and vocational education: A phenomenographic perspective. *High. Educ.* **2017**, *73*, 691–707. [[CrossRef](#)]
5. Cerda-Navarro, A.; Salva-Mut, F.; Comas Forgas, R. A typology of students in intermediate vocational education and training programmes based on student engagement factors, sociodemographic characteristics and intentions of dropping out. *Eur. J. Educ.* **2019**, *54*, 635–650. [[CrossRef](#)]
6. Cerdà-Navarro, A.; Quintana-Murci, E.; Salvà-Mut, F. Reasons for dropping out of intermediate vocational education and training in Spain: The influence of sociodemographic characteristics and academic background. *J. Vocat. Educ. Train.* **2022**, 1–25 [[CrossRef](#)]
7. Cordoba, C. Fundamentos del pensamiento en diseño. *Rev. Investig. Cienc. Soc. Hum.* **2015**, *6*, 38–50.
8. Latorre-Coscolluela, C.; Vázquez-Toledo, S.; Rodríguez-Martínez, A.; Liesa-Orús, M. Design Thinking: Creatividad y pensamiento crítico en la universidad. *Rev. Electrón. Investig. Educ.* **2020**, *22*, e28. [[CrossRef](#)]
9. Krüger, M. Design Thinking for German Vocational Schools? Discovering of an Innovative Approach by Testing in Teacher Education. *Open Educ. Stud.* **2019**, *1*, 209–219. [[CrossRef](#)]
10. Zhao, Y.; Ko, Y.O.J. Design thinking as an innovative pedagogy and innovation as a learning process in leading vocational education change in China: Lessons from two cases. In Proceedings of the 31st International Congress of School Effectiveness and School Improvement (ICSEI): Deepening School Change for Scaling: Principles, Pathways and Partnerships, Singapore, 8–12 January 2018; Nanyang Technological University: Singapore, 2018.
11. Woraphiphat, I.; Roopsuwankun, P. The impact of online design thinking-based learning on entrepreneurial intention: The case of vocational college. *J. Innov. Entrep.* **2023**, *12*, 2192–5372. [[CrossRef](#)]
12. Anwar, S.; Bascou, N.A.; Menekse, M.; Kardgar, A. A Systematic Review of Studies on Educational Robotics. *Journal of Pre-College Engineering Education Research. J. Pre-Coll. Eng. Educ. Res. (J-PEER)* **2019**, *9*, 2. [[CrossRef](#)]
13. Wing, J.M. Computational thinking. *Commun. ACM* **2006**, *49*, 33–35. [[CrossRef](#)]
14. Wing, J. Computational thinking and thinking about computing. In Proceedings of the 2008 IEEE International Symposium on Parallel and Distributed Processing, Miami, FL, USA, 14–18 April 2008; p. 1. [[CrossRef](#)]
15. Barr, V.; Stephenson, C. Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community? *ACM Inroads* **2011**, *2*, 48–54. [[CrossRef](#)]
16. Lee, I.; Martin, F.; Denner, J.; Coulter, B.; Allan, W.; Erickson, J.; Malyn-Smith, J.; Werner, L. Computational Thinking for Youth in Practice. *ACM Inroads* **2011**, *2*, 32–37. [[CrossRef](#)]
17. Grover, S.; Pea, R. Computational Thinking in K-12: A Review of the State of the Field. *Educ. Res.* **2013**, *42*, 38–43. [[CrossRef](#)]
18. Curzon, P.; Dorling, M.; Ng, T.; Selby, C.; Woollard, J. *Developing Computational Thinking in the Classroom: A Framework*; Project Report; University of Southampton: Southampton, UK, 2014.
19. Ioannou, I.; Angeli, C. A Framework and an Instructional Design Model for the Development of Students' Computational and Algorithmic Thinking. In Proceedings of the Mediterranean Conference on Information Systems, Paphos, Cyprus, 4–6 September 2016.
20. Bocconi, S.; Chiocciariello, A.; Dettori, G.; Ferrari, A.; Engelhardt, K. *Developing Computational Thinking in Compulsory Education—Implications for Policy and Practice*; JRC Research Reports JRC104188; Joint Research Centre: Seville, Spain, 2016.
21. Yadav, A.; Good, J.; Voogt, J.; Fisser, P. Computational Thinking as an Emerging Competence Domain. In *Competence-Based Vocational and Professional Education: Bridging the Worlds of Work and Education*; Mulder, M., Ed.; Springer International Publishing: Cham, Switzerland 2017; pp. 1051–1067. [[CrossRef](#)]
22. Su, J.; Yang, W. A systematic review of integrating computational thinking in early childhood education. *Comput. Educ. Open* **2023**, *4*, 100122. [[CrossRef](#)]
23. Lv, L.; Zhong, B.; Liu, X. A literature review on the empirical studies of the integration of mathematics and computational thinking. *Educ. Inf. Technol.* **2023**, *28*, 8171–8193. [[CrossRef](#)]
24. Chondrogiannis, E.; Symeonaki, E.; Papachristos, D.; Loukatos, D.; Arvanitis, K.G. Computational Thinking and STEM in Agriculture Vocational Training: A Case Study in a Greek Vocational Education Institution. *Eur. J. Investig. Health Psychol. Educ.* **2021**, *11*, 230–250. [[CrossRef](#)]
25. Díaz-Lauzurica, B.; Moreno-Salinas, D. Computational Thinking and Robotics: A Teaching Experience in Compulsory Secondary Education with Students with High Degree of Apathy and Demotivation. *Sustainability* **2019**, *11*, 5109. [[CrossRef](#)]
26. Tabet, N.; Gedawy, H.; Alshikhabobakr, H.; Razak, S. From Alice to Python. Introducing Text-Based Programming in Middle Schools. In Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education, (ITiCSE '16), New York, NY, USA, 11–13 July 2016; pp. 124–129. [[CrossRef](#)]

27. Dwyer, H.; Hill, C.; Hansen, A.; Iveland, A.; Franklin, D.; Harlow, D. Fourth Grade Students Reading Block-Based Programs: Predictions, Visual Cues, and Affordances. In Proceedings of the Eleventh Annual International Conference on International Computing Education Research, (ICER '15), New York, NY, USA, 18–20 June 2015; pp. 111–119. [[CrossRef](#)]
28. Papadakis, S.; Kalogiannakis, M.; Orfanakis, V.; Zaranis, N. The Appropriateness of Scratch and App Inventor as Educational Environments for Teaching Introductory Programming in Primary and Secondary Education. *Int. J. Web-Based Learn. Teach. Technol. (IJWLTT)* **2017**, *12*, 58–77. [[CrossRef](#)]
29. Papadakis, S.; Orfanakis, V. Comparing novice programming environments for use in secondary education: App Inventor for Android vs. Alice. *Int. J. Technol. Enhanc. Learn.* **2017**, *10*, 44–72. [[CrossRef](#)]
30. Papadakis, S.; Orfanakis, V. The Combined Use of Lego Mindstorms NXT and App Inventor for Teaching Novice Programmers. In Proceedings of the Educational Robotics in the Makers Era, Athens, Greece, 25 November 2017; Springer: Cham, Switzerland, 2017; pp. 193–204.
31. Aris, N.; Orcos, L. Educational Robotics in the Stage of Secondary Education: Empirical Study on Motivation and STEM Skills. *Educ. Sci.* **2019**, *9*, 73. [[CrossRef](#)]
32. Blumenfeld, P.C.; Soloway, E.; Marx, R.W.; Krajcik, J.S.; Guzdial, M.; Palincsar, A. Motivating Project-Based Learning: Sustaining the Doing, Supporting the Learning. *Educ. Psychol.* **1991**, *26*, 369–398. [[CrossRef](#)]
33. Strobel, J.; van Barneveld, A. When is PBL More Effective? A Meta-synthesis of Meta-analyses Comparing PBL to Conventional Classrooms. *Interdiscip. J. Probl. Based Learn.* **2009**, *3*, 44–58. [[CrossRef](#)]
34. Brown, A.L.; Campione, J.C. *Psychological Theory and the Design of Innovative Learning Environments: On Procedures, Principles, and Systems*; Lawrence Erlbaum Associates, Inc: Hillsdale, NJ, USA, 1996.
35. Mills, J.E.; Treagust, D.F. Engineering education: Is problem-based or project-based learning the answer? *Austral. J. Eng. Educ.* **2003**, *3*, 2–16.
36. Merritt, J.; Lee, M.; Rillero, P.; Kinach, B.M. Problem-Based Learning in K–8 Mathematics and Science Education: A Literature Review. *Interdiscip. J. Probl. Based Learn.* **2017**, *11*. [[CrossRef](#)]
37. Kelley, K.W.; Fowlin, J.M.; Tawfik, A.A.; Anderson, M.C. The Role of Using Formative Assessments in Problem-based Learning: A Health Sciences Education Perspective. *Interdiscip. J. Probl. Based Learn.* **2019**, *13*. [[CrossRef](#)]
38. Ulger, K. The Effect of Problem-Based Learning on the Creative Thinking and Critical Thinking Disposition of Students in Visual Arts Education. *Interdiscip. J. Probl. Based Learn.* **2018**, *12*. [[CrossRef](#)]
39. Jalinus, N.; Nabawi, R.A.; Mardin, A. The Seven Steps of Project Based Learning Model to Enhance Productive Competences of Vocational Students. In Proceedings of the International Conference on Technology and Vocational Teachers (ICTVT 2017), Yogyakarta, Indonesia, 28 September 2017; Atlantis Press: Amsterdam, The Netherlands, 2017; pp. 251–256. [[CrossRef](#)]
40. Chiang, C.L.; Lee, H. The Effect of Project-Based Learning on Learning Motivation and Problem-Solving Ability of Vocational High School Students. *Int. J. Inf. Educ. Technol.* **2016**, *6*, 709–712. [[CrossRef](#)]
41. Martín de Soto Domínguez, J. Formación Profesional Básica: La etapa educativa al servicio de la equidad. *Padres Maest./J. Parents Teach.* **2021**, *385*, 26–31. [[CrossRef](#)]
42. Thomas, J. *A Review of Research on Project-Based Learning*; The Autodesk: San Rafael, CA, USA, 2000.
43. Megayanti, T.; Busono, T.; Maknun, J. Project-based learning efficacy in vocational education: Literature review. In Proceedings of the IOP Conference Series: Materials Science and Engineering, Chennai, India, 16–17 September 2020; IOP Publishing Ltd.: Bristol, UK, 2020; Volume 830. [[CrossRef](#)]
44. Guan, L.; Yuan, N.; Xie, Y.; Feng, J.; Sun, B.; Li, L. A study on the application of flipped classroom teaching in higher vocational education. In Proceedings of the 2015 10th International Conference on Computer Science & Education (ICCSE), Cambridge, UK, 22–24 June 2015; pp. 819–823. [[CrossRef](#)]
45. Villalba, M.T.; Castilla, G.; Redondo, S. Factors with Influence on the Adoption of the Flipped Classroom Model in Technical and Vocational Education. *J. Inf. Technol. Educ. Res.* **2018**, *17*, 441–469. [[CrossRef](#)]
46. Olivos, P.; Santos, A.; Martín, S.; Cañas, M.; Gómez-Lázaro, E.; Maya, Y. The relationship between learning styles and motivation to transfer of learning in a vocational training programme. *Suma Psicol.* **2016**, *23*, 25–32. [[CrossRef](#)]
47. Kholifah, N.; Sudira, P.; Rachmadtullah, R.; Nurtanto, M.; Suyitno, S. The effectiveness of using blended learning models against vocational education student learning motivation. *Int. J. Adv. Trends Comput. Sci. Eng.* **2020**, *9*, 7964–7968.
48. Jayalath, J.; Esichaikul, V. Gamification to Enhance Motivation and Engagement in Blended eLearning for Technical and Vocational Education and Training. *Technol. Knowl. Learn.* **2022**, *27*, 91–118. [[CrossRef](#)]
49. Lihui Sun, Z.G.; Hu, L. Educational games promote the development of students' computational thinking: A meta-analytic review. *Interact. Learn. Environ.* **2023**, *31*, 3476–3490. [[CrossRef](#)]
50. Bacca, J.; Baldiris, S.; Fabregat, R.; Kinshuk; Graf, S. Mobile Augmented Reality in Vocational Education and Training. *Procedia Comput. Sci.* **2015**, *75*, 49–58. [[CrossRef](#)]
51. Bacca Acosta, J.L.; Baldiris Navarro, S.M.; Fabregat Gesa, R.; Kinshuk, K. Framework for designing motivational augmented reality applications in vocational education and training. *Austral. J. Educ. Technol.* **2019**, *35*. [[CrossRef](#)]
52. Angraini, L.M.; Yolanda, F.; Muhammad, I. Augmented reality: The improvement of computational thinking based on students' initial mathematical ability. *Int. J. Instr.* **2023**, *16*, 1033–1054. [[CrossRef](#)]
53. Boonk, L.M.; Gijssels, H.J.M.; Ritzen, H.; Brand-Gruwel, S. Student-perceived parental involvement as a predictor for academic motivation in vocational education and training (VET). *J. Vocat. Educ. Train.* **2022**, *74*, 187–209. [[CrossRef](#)]

54. Yazan, B. Three approaches to case study methods in education: Yin, Merriam, and Stake. *Qual. Rep.* **2015**, *20*, 134–152. [[CrossRef](#)]
55. Mitchell, J.C. Case and Situation Analysis. *Sociol. Rev.* **1983**, *31*, 187–211. [[CrossRef](#)]
56. Cross, N. *Design Thinking: Understanding How Designers Think and Work*; Bloomsbury Publishing: London, UK, 2023.
57. Guaman-Quintanilla, S.; Everaert, P.; Chiluiza, K.; Valcke, M. Impact of design thinking in higher education: A multi-actor perspective on problem solving and creativity. *Int. J. Technol. Des. Educ.* **2023**, *33*, 217–240. [[CrossRef](#)]
58. Calavia, M.B.; Blanco, T.; Casas, R.; Dieste, B. Making design thinking for education sustainable: Training preservice teachers to address practice challenges. *Think. Ski. Creat.* **2023**, *47*, 101199. [[CrossRef](#)]
59. Thom, R. A design Thinking Approach to Professional Development in Reasonable Adjustment: A New Methodology for Trainers in the Vocational Education and Training Sector of Western Australia. Master's Thesis, Edith Cowan University, Joondalup, Australia, 2016.
60. Harth, T.; Panke, S. Design Thinking in Teacher Education: Preparing Engineering Students for Teaching at Vocational Schools. *Int. J. Learn.* **2019**, *18*, 413–439.
61. Panke, S. Design Thinking in Education: Perspectives, Opportunities and Challenges. *Open Educ. Stud.* **2019**, *1*, 281–306. [[CrossRef](#)]
62. Selltitz, C.; Jahoda, M.; Deutsch, M.; Cook, S. *Research Methods in Social Relations*; Holt, Rinehart and Winston: New York, NY, USA, 1959.
63. Weitzman, E.; Miles, M. *Computer Programs for Qualitative Data Analysis*; SAGE Publications: London, UK, 1995.
64. Silver, C.; Lewins, A. *Using Software in Qualitative Research: A Step-by-Step Guide*; SAGE Publications: London, UK, 2014. [[CrossRef](#)]
65. Sampieri, H. *Methodology of the Research*; McGrawHill: New York, NY, USA, 2014.
66. Mertens, D. *Research and Evaluation in Education and Psychology: Integrating Diversity with Quantitative, Qualitative, and Mixed Methods*; SAGE Publications: London, UK, 2010.
67. Savin-Baden, M.; Major, C. *Qualitative Research: The Essential Guide to Theory and Practice*; Routledge: Abingdon, UK, 2013.
68. Moreno-Salinas, D.; Pascoal, A.M.; Aranda, J. Optimal Sensor Placement for Multiple Target Positioning with Range-Only Measurements in Two-Dimensional Scenarios. *Sensors* **2013**, *13*, 10674–10710. [[CrossRef](#)]
69. Moreno-Salinas, D.; Pascoal, A.; Aranda, J. Optimal Sensor Placement for Acoustic Underwater Target Positioning With Range-Only Measurements. *IEEE J. Ocean. Eng.* **2016**, *41*, 620–643. [[CrossRef](#)]
70. Blasco, C.T. Metodología de Enseñanza-Aprendizaje de los Lenguajes HTML y CSS para Estudiantes con Predominancia Lateral Derecha: El Caso de Estudiantes de Diseño Gráfico. Master's Thesis, Universidad de La Rioja, Logrono, Spain, 2016.
71. Grinnell Jr, R.M.; Unrau, Y. *Social Work Research and Evaluation: Quantitative and Qualitative Approaches*; Cengage Learning: Boston, MA, USA, 2005.
72. Nilsson, L.; Eklund, M.; Nyberg, P.; Thulesius, H. Driving to Learn in a Powered Wheelchair: The Process of Learning Joystick Use in People With Profound Cognitive Disabilities. *Am. J. Occup. Ther.* **2011**, *65*, 652–660. [[CrossRef](#)]
73. Somekh, B. *Action Research*; McGraw-Hill Education: London, UK, 2005.
74. Bradbury-Huang, H. What is good action research?: Why the resurgent interest? *Action Res.* **2010**, *8*, 93–109. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.