# Teaching computational thinking with a tangible development platform: An exploratory field study at school with Kniwwelino

Lou Schwartz[1] · Valérie Maquil[1] · Laurence Johannsen[1] · Christian Moll[1] ·
Johannes Hermen[1]

## Abstract

Computational Thinking (CT) is an emerging topic in school curricula. Different tools exist to support the learning of CT, namely visual programming languages and tangible development platforms (TDP), which are widely used in extra-curricular activities. To date, few tools have been developed that consider both teachers' needs and the school context. We designed the Kniwwelino Classroom Kit (KCK) in order to support the teaching of CT in classrooms. This paper presents the results of a pilot study, in which the KCK was used in three primary and three secondary schools in Luxembourg. The KCK is based on Kniwwelino, a creative environment designed for children from 8 years old to learn about programming and electronics. The study explored the suitability of the KCK for teaching CT in primary and secondary schools, focusing on three research questions: (1) how the KCK supports teachers' pedagogical practices, (2) how the KCK fulfils the pedagogical objectives set by teachers, and (3) the user experience of the KCK (for teachers and pupils). To assess how the KCK meets the three above-mentioned objectives, we used a mixed method approach, combining semi-guided interviews, logbooks, and several questionnaires. We learned that the KCK is versatile enough to adapt to different environments, objectives, and pedagogical approaches, which responds to the first two research questions, and thus offers a great deal of freedom to teachers. The fun and tangible approach of the KCK introduces programming and enables pupils to develop both CT and soft skills such as communication and collaboration. Teachers' feedback on their user experience was positive, particularly concerning stimulation, novelty, and attractiveness. KCK was evaluated as interesting, valuable, and good, but it was noted that some improvements needed to be made to make it easier, more predictable, and faster. The pupils also rated their user experience positively, with values that were generally on a par with previous studies reported in the state of the art regarding the use of Scratch and BBC Micro:bit in school-based experiments for learning CT. Drawing from these learnings, we list several recommendations for the development of other similar TDP tools to support the teaching of CT in a classroom

---

Extended author information available on the last page of the article

setting. These recommendations concern the material, programming interface, connection, educational material, and the creation of a community.

Nowadays, understanding computers is essential for living in our information-driven society and computer-related skills have become a prerequisite for any job. Teaching computational thinking is therefore increasingly important. Computational thinking (CT) is defined by Wing as "*the thought processes involved in formulating a problem and expressing its solution(s) in such a way that a computer—human or machine—can effectively carry out.*" (Wing, 2017). Angeli defines five elements that make up computational thinking: **abstraction** (deciding what information about an entity/object to keep and what to ignore), **generalization** (formulating a solution in generic terms so that it can be applied to different problems), **decomposition** (breaking down a complex problem into smaller parts that are easier to understand and solve), **algorithms** (devising a step-by-step set of operations/actions in order to solve a problem), and **debugging** (identifying, removing, and fixing errors) (Angeli et al., 2016).

As CT is an emerging teaching subject, several kinds of tools were developed to support teachers in this task. Among the most promising tools are visual programming languages and tangible development platforms (TDP). This paper focuses on the latter, and particularly on the Kniwwelino platform, which is a Luxembourgish initiative. The aim of this paper is to better understand how a TDP can serve teachers and pupils when teaching CT in the classroom, and what unique opportunities a TDP can provide.

This paper begins with a state of the art concerning the teaching of CT, including the tools that currently exist to support teachers, and studies done on their use in classroom settings. Thereafter, we will introduce the study conducted in six different schools in Luxembourg during the 2019–2020 school year using the Kniwwelino Classroom Kit. The results of this study are presented in the third part of this paper and are discussed in part 4 to state the lessons learned.

# 1 State of the art: Teaching computational thinking in the classroom

In this section, we list the different kind of tools usable for CT teaching. Then, we focus on the Tangible Development Platforms (TDP). Finally, we present studies on how these tools are used in classrooms to support CT teaching.

## 1.1 Tools to support the teaching of computational thinking

To date, different types of tools have been proposed to support CT learning. Ching et al. listed the following kinds (Ching et al., 2018): programming toys (such as

Think & Learn Code-a-pillar™), robot kits (such as LEGO® Mindstorms EV3), board games (such as Robot Turtles Game), augmented reality tools (such as Osmo Coding Awbie), programming concept-practicing applications or websites (such as code.org), and animation/game development tools (such as Scratch). To this list, we can also add TDPs, which connect to electronic extensions and on-screen programs, such as BBC Micro∶bit or Kniwwelino. They are as powerful and versatile as the animation development tools, but in addition, allow users to learn about electronics and are tangible.

Visual programming languages are the most widely used tools for teaching CT (Hsu et al., 2018). Lye and Koh recommend the use of visual programming languages to teach CT to primary and secondary school children (K-12), to allow them to focus on logic rather than syntax (Lye & Koh, 2014). Kalogiannakis et al. also note that visual programming reduces writing problems compared to a text-based programming language due to the shape of the blocks, which guides pupils to combine them (Kalogiannakis et al., 2021). However, this does not help them better understand the underlying concepts and the procedure.

The use of microcontrollers associated with visual programming makes it possible to play on the visual and tangible aspects of the learning tools. This makes it easier to integrate CT thanks to the concrete visualization of the effects of the program "*written*" by the pupil, especially for the test and debugging dimensions (Lye & Koh, 2014; Scherer et al., 2020). In addition, the possibility of immediate and tangible feedback in the real world seems to be beneficial for CT learning by strongly influencing motivation and engagement (Scherer et al., 2020). By designing games and projects, the pupil receives insights into the design and implementation of applications in an engaging and fun way (Scherer et al., 2020). Kastner-Haulner et al. pointed out that visual programming, i.e., block-based, combined with physical computing, i.e., a TDP, can foster the CT skills of primary school (Kastner-Hauler et al., 2022).

### 1.2 Tangible development platforms (TDP) to teach computer thinking

Lately, affordable TDPs have emerged: Arduino (Arduino, 2022), BBC Micro:bit (BBC, 2022), Calliope mini (Calliope, 2022), Lego Mindstorms (Lego, 2022), littleBits (Bdeir, 2009), MakeMe (Johnson et al., 2016), Talkoo (Katterfeldt et al., 2018), and Kniwwelino (LIST, 2022; Maquil et al., 2018). A TDP is composed of a programming platform that enables users to compose a program sent on a tangible object, often a microcontroller board. The physical computing device provides the link between abstract concepts and tangible experiences. The aim of these platforms is to offer a simple, fun and engaging way to make and program electronic objects to develop children's CT skills.

### 1.3 Studies on teaching computational thinking in the classroom

The most widely used visual programming platform for teaching CT is Scratch (Hsu et al., 2018). Its use for introducing programming to children has been evaluated in

several studies. Sáez et al. validate the relevance of learning programming with a visual programming platform (Scratch) with primary school pupils (fifth and sixth grade) in a study conducted with 107 pupils in science and art classes in five different schools in Spain over a period of two years (Sáez-López et al., 2016). The use of Scratch resulted in the increased motivation, enjoyment, engagement, and enthusiasm of the pupils, indicating improvements related to CT and computing practices.

In Kaučič's study, learning to program with Scratch was evaluated with eleven fifth grade pupils, nine eighth grade pupils, and sixteen second grade pupils completing a five-month introductory programming curriculum with Scratch (Kaučič & Asič, 2011). The training consisted of an introduction to Scratch, learning-by-doing through experimentation, guided lessons with the introduction of new content, group work on examples, and individual task-solving with an emphasis on problem-solving. In this study, Scratch was often the participants' first experience of programming. Following this use of Scratch in the classroom, pupils were asked to complete a questionnaire with a five-point Likert scale (with 1 corresponding to total disagreement with the statement, and 5 corresponding to total agreement). The answers to the questionnaire were as follows: Scratch is attractive (I like it) (3.81), Scratch was easy to use (3.94), Scratch was interesting for usage (4.22), Scratch was motivating me for work (3.25), Scratch was positive experience for me (4.06), Scratch was fun to work with (4.03), Scratch is suitable for learning programming (4.08), Programming is hard (3.31), and I still know how to use Scratch (3.08).

A study conducted by Shina shows that 60% of the pupils surveyed found it easy to use Scratch (Schina et al., 2019). The study consisted of 27 children aged 9 to 12 participating in a one-and-a-half-hour introductory Scratch workshop at a non-formal education institution.

As yet, few works exist reporting on the use of TDP to teach CT. One of these few examples is Sentance et al., which studied the use of the BBC Micro:bit board to teach CT in schools in the UK (Sentance et al., 2017). The study found that the tangible aspect of the board increased motivation and encouraged collaboration and creativity, and that it gave pupils a better understanding of what was going on in the program.

Kalogiannakis et al. undertook a review of studies using BBC Micro:bit (Kalogiannakis et al., 2021). This review notes that most teachers first used the official activities, although some of them then developed their own pedagogical material. The authors argue that this is certainly due to a lack of confidence on the part of the teachers. Concerning motivation, high motivation among the pupils was observed at the beginning of the use, however, it decreased over time. It also seems that highly capable students are less motivated because they feel limited by the device's capabilities. In terms of the visual programming interface, some issues with BBC Micro:bit were observed when using online compilers, which were sometimes difficult to access where the connection was unreliable.

Overall, these and other studies have reported some positive learning effects with different platforms, however, they essentially focus on the impact on the pupils. The objective of this paper is to investigate the opportunities and limitations when using TDPs in a real classroom setting, considering the needs of both teachers and pupils. How can TDPs be used in a school context to teach programming and CT? Do

TDPs address the needs of teachers and pupils in different settings? These questions will be discussed in this paper on the basis of a pilot study that used Kniwwelino to teach CT in primary and secondary schools conducted with several schools in Luxembourg.

## 2 Field study in Luxembourg with Kniwwelino

In order to understand how a TDP can be used in a school setting for teaching CT, we conducted an exploratory field study with 10 primary and secondary school classes in six schools in Luxembourg. This study took place between September 2019 and July 2020, which represents one school year. After introducing the particular context of teaching CT in the Luxembourg education system, we presented the Kniwwelino Classroom Kit, and explained how we had adapted this TDP for use by teachers in a classroom setting. Then, we described the research questions that guided this study, followed by the settings of the study: its participants, the indicators chosen, and the testing procedure, which included the pedagogical objectives, pedagogical practices, and organization of each teacher. The results obtained are discussed in the following section.

### 2.1 Luxembourgish context: Teaching computer thinking in Luxembourg

CT was first introduced into the Luxembourgish education system in September 2020 (i.e., after the study presented in this paper) in cycle 4 of the primary school (pupils aged 10 to 12 years old), and has been progressively extended to cycles 1, 2 and 3 of the primary school (students aged 4 to 9 years old) as well as to secondary schools, with the launch of a new course in digital science in September 2021.

The Digital(4)Education initiative, which considers ICT both a pedagogical tool and a taught subject, consists of the introduction of coding in cycle 4 classes (ages 10–12 years) and began in September 2020. The activities proposed aim to prepare students to understand the digital world from an early age and to master the basic skills essential for tomorrow's professions. The circular of spring 2021 refers to the creation of fifteen positions for teachers specialized in digital skills (MEN, 2021). These teachers took up their duties at the start of the 2020–2021 school year to support other teachers in each region with the introduction of coding in the primary school curriculum. Programming skills are taught together with mathematical skills or across different subjects such as languages, science awareness, or creative awareness. The following school year, coding was introduced in cycles 1 to 3 (ages 4–9 years), also in a cross-curricular manner.

Therefore, when we carried out the present study, the teaching of CT was not yet part of the official curriculum, but teachers had been informed that there was a directive on the subject that would require it to be added to the national curriculum in primary and secondary schools in the coming years.

## 2.2 Material used: The Kniwwelino Classroom Kit (KCK)

Kniwwelino® is a TDP developed in Luxembourg that was popular with children during maker fairs. Documentation for Kniwwelino is available in German, French, and English, the main languages used in classrooms in Luxembourg. Kniwwelino had been identified by teachers during these fairs as a tool for teaching CT. For these reasons, we chose to use it as the basis of our work. In this section, we present Kniwwelino and describe how it was adapted with the objective of supporting CT teaching specifically in the classroom setting.

Kniwwelino is a creative environment designed for children from 8 years old to learn about basic programming and electronics (Maquil et al., 2018). The Kniwwelino environment includes an electronic board based on Arduino® and Wi-FiTM technologies, and a fun, visual online programming interface where blocks are aggregated as in a puzzle.

The Kniwwelino hardware is designed around the Espressif ESP8266 microcontroller-based SoC ESP-12 Wi-Fi module, which is feature-rich but with a small footprint. Besides the previously mentioned Wi-Fi functionality, it comes with 15 general purpose input/output (GPIO) and 1 analog-to-digital convertor (ADC) pin. For debugging and firmware uploads, a serial interface is also available. The printed circuit board (PCB) design was kept small and round to make it easy to integrate it into individual hardware projects. For interaction and feedback, two buttons, a $5 \times 5$ LED (light-emitting diode) matrix, and one RGB LED (mixing red, green, and blue sources LED) are placed on the front, surrounded by an alligator clip and sewing-friendly GPIO pads, while all other electronics are placed on the back. The number of GPIO pads was limited to five. This was done to expose only the pads that have no dual function and to not overwhelm the targeted audience with too many options. Extensions, connected to the board, allow users to discover electronics.

The interest of Kniwwelino, compared to BBC Micro:bit for example, is notably its Wi-Fi connection and the use of MQTT (MQ Telemetry Transport) to exchange messages between the different boards or with other connected objects or web services (MQTT, 2022). MQTT is a Client Server publish/subscribe messaging transport protocol. MQTT is helpful for the integration and processing of data collected by Kniwwelino or external sources from the internet, especially in combination with an event-driven flow-based programming environment called Node-RED (Foundation & Contributors, 2022), which is perfectly adapted to receiving and processing MQTT messages. It can be attached to the MQTT broker of Kniwwelino in a decentralized approach and can act as a data provider, e.g., Node-RED pre-processes weather forecast data from an external service for the Kniwwelino platform.

### 2.2.1 Adaptation to the classroom environment

Given the success of Kniwwelino with children and their supervisors at eight national fairs for children to discover CT, coding, making, and do-it-yourself, such as MakerFests, Researchers' Day, and the Science Festival (each of these events bring together more than 2,000 participants), as well as two workshops with children to explore CT with Kniwwelino and other events in Luxembourg, we decided

to offer a version adapted to classroom use. In order to do this, we deployed a user-centered approach. We first interviewed teachers individually to collect their actual practices and their intentions in using TDP in their classrooms. We then organized design workshops with teachers, where after learning to use Kniwwelino, they imagined and shared the kind of projects they would like to develop with Kniwwelino in their classrooms (see Fig. 1). We were then able to propose the first version of the Kniwwelino Classroom Kit (KCK).

The KCK includes boxes for pupils to work in pairs containing: an **electronic board** and its shield, connection wires, and numerous **electronic extensions** (3 LEDs, a Neopixel LED strip, a servo motor, a button, a buzzer, a temperature sensor, a distance sensor, a sound sensor, a light sensor, a potentiometer, and alligator clips); see Fig. 2.

In addition, the KCK contained **educational materials** consisting of a user manual for teachers, ready-to-use activity sheets for progressively learning the basics of programming and electronics, and step-by-step ideas for cross-curricular technology projects (e.g., a weather station). It also contained **creative materials** to challenge pupils' creativity and problem-solving capacities: mission cards, extension cards, a template for the project description sheet, and dices, see Fig. 3. Mission cards
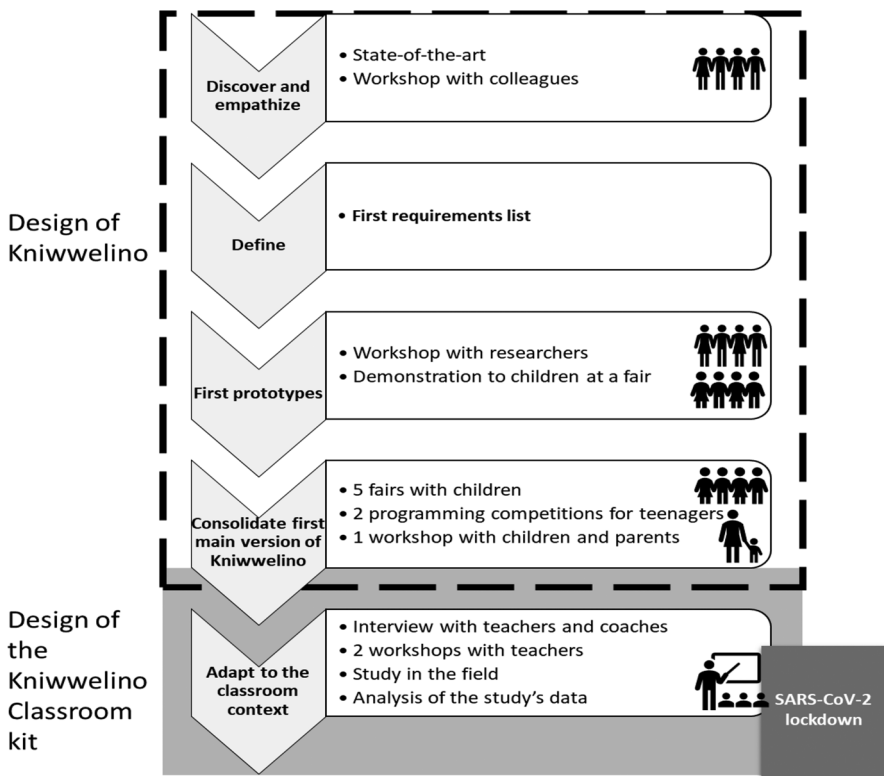


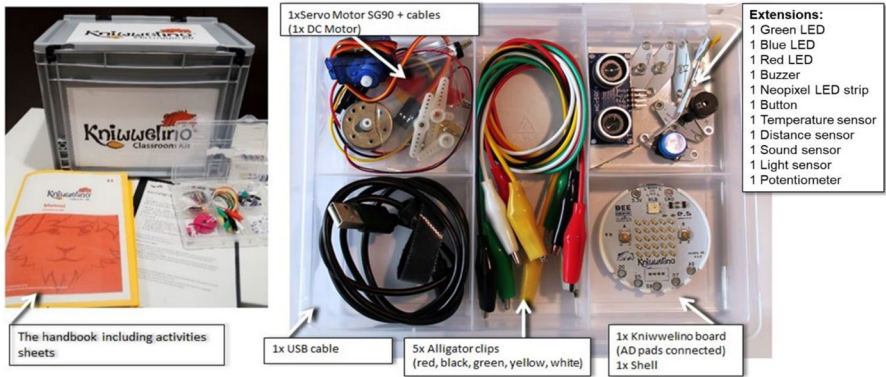**Fig. 1** The Kniwwelino Classroom Kit Design

**Fig. 2** The Kniwwelino Classroom Kit and its contents



**Fig. 3** Pedagogical materials: an activity sheet, a project sheet, 3 extension cards, dices, and 3 mission cards

propose challenges for pupils to start developing their own Kniwwelino project. Extension cards explain how to connect each extension and which blocks can be used to program it into the visual programming interface. The dices include one dice indicating the different missions and two dices indicating the different extensions. The pupils roll the dices and have to come up with an idea to solve the challenge with the two extensions shown by the dice. The educational and creative materials were all available in printed, updatable (in Word and PowerPoint) formats, and online on a wiki.

### 2.2.2 Equipment provided

Each pilot school received a KCK, containing 15 to 20 pupil boxes (see Fig. 2) and a booklet containing educational and creative materials (see Fig. 3). It was also possible to borrow a 4G router, an option that the pilot schools did not choose at the beginning of the study.

## 2.3　Research questions

This study focuses on the use of the KCK by teachers and seeks to answer the following research question: **How suitable is the KCK for teaching CT in primary and secondary schools?** It is divided into three sub-questions:

- Q1: In what way does the KCK support teachers' pedagogical practices?
- Q2: How does the KCK fulfill the pedagogical objectives set by teachers?
- Q3: What is the user experience of the KCK (for teachers and pupils)?

## 2.4　Study participants

### 2.4.1　Selection of pilot classes

The present study was conducted in six schools, selected to represent the diversity of the educational sector in Luxembourg, covering primary and secondary education, public and private schools, teaching in different languages (specific to Luxembourg), and geographical location throughout the country; see the summary in Table 1.

### 2.4.2　Profiles of teachers who participated in the study

Eleven teachers (7 male and 4 female) participated in the study on a voluntary basis. The participating teachers had experience in teaching CT, except at **P1** and one of the teachers at **P3**. However, at **P3**, a digital learning coach supported the teachers. The following data is based on the nine teachers who completed their profile. Regarding the seniority of these nine teachers: 4 teachers have more than 20 years of seniority, 2 have between 10 and 20 years of seniority and 2 have less than 5 years of seniority. Before using Kniwwelino, regarding their computer skills: 4 teachers felt they had an intermediate level, 3 felt they were competent and 2 felt they were experts (scale used: novice, intermediate, competent, expert). Regarding programming before using Kniwwelino: 3 teachers declared themselves novices, 3 had an intermediate level, 1 felt competent and 2 declared themselves experts. The tools cited as having been used to teach CT and programming were (in brackets is the number of occurrences): Scratch (6), Kniwwelino (5), Lego (4), code.org (4), Java

**Table 1**　Pilot schools that took part in the KCK field study

| School | Education | Type | Location | Language | Number of pupils | Age |
|---|---|---|---|---|---|---|
| P1 | primary | public | south | French | 25 (1 class) | 9–11 |
| P2 | primary | public | capital | Luxembourgish | 8 (1 class) high-capable pupils | 10–12 |
| P3 | primary | private | capital | English | 90 (5 classes of 18 pupils | 9–11 |
| S1 | secondary | public | south | Luxembourgish | 26 (2 groups of 13 pupils) | 14–15 |
| S2 | secondary | public | capital | English, Luxembourgish | 44 (2 classes of 22 pupils) | 12–13 |
| S3 | secondary | private | capital | French | / | / |

(3), Arduino (2), Logo (2), Python (2), MySQL (1), Raspberry Pi (1), and BBC Micro:bit (1).

### 2.4.3 Profiles of pupils who participated in the study

138 pupils (57 female and 81 male, average age: 10.19 from 8 to 16 years old) from the six pilot schools responded to our pre-use questionnaire. Most respondents said they had programmed before (76 of 137 respondents, or 55.47%), mainly at school (58 respondents) or at home (33 respondents). Pupils think programming is fun with an average score of 4.47 (min 1; max 5) and most of them would like to become a developer (average 3.85 with min 1; max 5). There is a small difference between boys and girls, but nothing significant (for the question "*it's fun to program*" the average was 4.47 for boys and 4.37 for girls; and for the question "*become a developer*" the average was 3.61 for boys and 3.53 for girls). It is important to note that the students that participated in the study were not pre-selected and that their participation was not voluntary, meaning that whole classes participated in the study.

## 2.5 Indicators and data collection methods

To collect the data for this study, we used three questionnaires, a logbook, and a semi-structured interview, as detailed in the following sections.

### 2.5.1 Logbook

Teachers were given a logbook to complete at the end of each session, in which they described the session (types of activities carried out, learning objectives, educational materials, and extensions used, etc.) and indicated their observations.

### 2.5.2 Interview with teachers

After the period of classroom use of the KCK, we conducted a semi-structured interview with each teacher. This interview allowed us to collect qualitative data on their objectives and the pedagogical approach used, the actual course of the sessions and learning, the impact of the use of the KCK on the pupils, their opinion of the KCK, and their perspectives in terms of reuse and transfer of the acquired skills.

### 2.5.3 User experience questionnaire for teachers

To measure the user experience provided by the KCK, we asked teachers to fill out the User Experience Questionnaire (UEQ) after using the KCK in the classroom (Laugwitz et al., 2008). This questionnaire measures attractiveness, perspicuity, efficiency, dependability, stimulation, and novelty. For each scale, pairs of opposing words were presented. The respondent was asked to indicate which of these two words corresponded more closely to his/her feelings using a scale from -3 to +3, including 0, which was neutral.

### 2.5.4 Pre-use and post-use questionnaires for pupils

We asked pupils to complete two questionnaires: the first (pre-use questionnaire) to be completed before the KCK presentation, the second (post-use questionnaire) to be completed after the training provided by their teachers. The pupils answered both questionnaires on a five-item Likert scale using smiley emoticons as recommended for children aged 9–11 years old by (Hall et al., 2016). The questionnaires aimed to measure the pupils' attitudes towards programming in general and their user experience of the KCK. The pre-use questionnaire also provided insights into the demographic distribution of the study participants.

## 2.6 Test procedure

Before conducting the experiment, we invited interested teachers from the pilot schools to two introductory training workshops of four hours each. The first workshop was an introduction to Kniwwelino with exercises, followed by a presentation of the educational materials provided, with the aim of learning how to use them. The teachers were each given a pupil box to try out Kniwwelino in their school or home.

The second workshop focused on teachers' project ideas for their pupils, with exchanges between participants and trainers to make the project feasible and collect feedback from teachers on improvements to the pupil box. These workshops took place five to seven months before the use of the tool in the field, to give the teachers time to appropriate the tool and to refine their pedagogical concept.

Of the nine teachers who responded to the questionnaire, two teachers attended only the first workshop, two teachers attended only the second workshop, three teachers attended both workshops, and the two teachers who were experts in computer science and programming did not attend the workshops.

In these workshops, we also explained the role of the pilot school, which was to: (1) commit to delivering at least three programming sessions to their pupils, with teachers free to choose the format and content of the sessions; (2) return the evaluation documents needed for the study.

1. The testing procedure consisted of three steps, summarized in Fig. 4:
2. At the beginning of the first session: ask pupils to complete the pre-questionnaire.
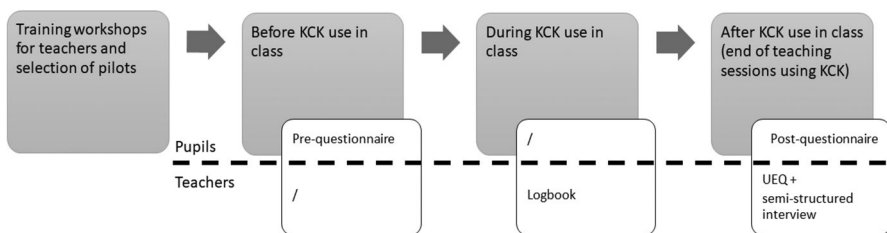3. At the end of each session: complete the logbook.



**Fig. 4** Study protocol

4.  At the end of the period of use of the KCK in the classroom: fill in the UEQ and ask the pupils to fill in the post-use questionnaire. In addition, teachers were invited to share their experience in a semi-structured interview.

## 2.7  Analysis of collected data

After a transcription phase for the interviews, the logbooks and the interviews were analyzed using the thematic analysis method (Boyatzis, 1998). There was then a familiarization phase, after which three researchers analyzed the various qualitative data collected separately to identify the main topics identified in the teachers' feedback. Then, they reviewed the themes identified and discussed their analysis together. At the end, one researcher oversaw the synthesis of the analyses.

The different questionnaires were analyzed using statistics. For the UEQ questionnaire, the official Data Analysis Tool was used.

## 2.8  Ethical and legal issues

As the study took place at school, not on a voluntary basis, and with participants who were minors, it would have been difficult to get permission from all parents. So, to ensure equity and preserve the privacy of students, we decided that no researchers would be present during the learning sessions and that no photos would be taken, or videos made. We also took care to not ask for personal data through questionnaires distributed to the pupils.

## 2.9  Definition of the pilot studies

This section presents the pedagogical objectives defined by the teachers, the way they organized the learning sessions, and the pedagogical practices they used to achieve the objectives set.

### 2.9.1  Pedagogical objectives

All teachers had the objective of using the KCK primarily to introduce their pupils to CT and programming. However, they also wanted to use the KCK to address several sub-goals or complementary objectives, as shown in Table 2 (presenting only the answers of the nine teachers who answered). Quotations are given in the original language to illustrate and are translated if necessary:

- Ability to investigate a subject, solve problems, and acquire logical skills. "My learning objective was to see if we could get [an] 8-year-old to engage with [a] codable device in a classroom setting. I wanted to see if we could get computational thinking which is not actually coding but understanding what an algorithm or variable is." (**P3**).
- Introduction to programming to prepare pupils for the use of programming languages such as Python. "On utilise Kniwwelino blockly pour les cours inférieurs, en prépa-

**Table 2** Teaching objectives and achievements as reported by teachers (answers from 9 of the 11 teachers who responded). T means Teacher

| Teacher | P1-T1 | P2-T2 | P3-T3 | P3-T4 | P3-T5 | S1-T6 | S2-T7 | S2-T8 | S3-T9 |
|---|---|---|---|---|---|---|---|---|---|
| Objective | | | | | | | | | |
| CT | ■ | ■ | ■ | □ | ■ | ■ | ■ | ■ | / |
| Programming | ■ | ■ | ✱ | □ | □ | ■ | / | ■ | □ |
| Electronics | ■ | / | / | □ | / | □ | / | □ | / |
| Problem solving | ■ | ■ | ■ | □ | ■ | ■ | ■ | ■ | □ |
| Logic | ■ | ■ | □ | □ | ■ | □ | ■ | ■ | □ |
| Creativity | □ | ■ | ■ | □ | ■ | ■ | ■ | ■ | □ |
| Collaboration | ■ | □ | ■ | □ | ■ | ■ | / | ■ | □ |
| Communication | ■ | □ | ■ | □ | ■ | ■ | / | ■ | □ |
| Empowerment | / | ■ | ■ | □ | ■ | / | / | ■ | □ |
| Stimulate interest in IT careers | No | Yes | Yes | No | No | Yes | Yes | Yes | Yes |
| Lockdown management | Stop | Stop | Finished before | Stop | Finished before | Continued | Stop | Continued | Stop |

Caption:

■ objective achieved

□ objective not achieved

✱ was not an objective but was acquired

/ was not an objective

ration de notre cours de langage de programmation en Python. On fera le lien vers Kniwwelino ou Scratch qu'ils ont eu en 5°". (We use Kniwwelino blockly for the lower courses, in preparation for our programming language course in Python. We will relate this to [the] Kniwwelino or Scratch that they had in class 5) (S2).

- Discovering debugging.
- Introduction to electronics.
- Development of the ability to collaborate and help each other. "*One of the big parts* of this was that the kids were working together and having to communicate and help each other. (...) They had to share their knowledge with their partner and talk them through it. That's an added component to this, that's fantastic." (**P3**).
- Development of communication skills, especially in school **P2** where pupils were asked to document their work through presentations to their parents. In **S3**, the objective of acquiring communication skills was strong with the desire to have high school pupils pass on the use of the KCK to elementary pupils. **S3**.
- Gain in autonomy.
- Creativity through the realization of projects or experimentation with the available options.
- Get children interested in IT careers.

### 2.9.2 Organization of learning sessions

The individual schools organized their learning sessions differently, see Table 3. In **P1** and **P3** the pupils worked in pairs. In **S1** the pupils were separated into two groups: the first had instruction in the first semester that took place both in school and via distance learning at home, the second had instruction in the second semester and, because of the lockdown, participated only in distance learning at home. Pupils worked first on their own and then in groups to create their project. In **S2**, the classes were divided into two subgroups with a weekly session of one hour each, and within these subgroups, the pupils worked individually, as in school **P2**.

**Table 3** Organization of learning sessions and materials used by pilot schools

| Pilot | Alone or in pairs | Number of sessions | Educational material used | Materials used |
|---|---|---|---|---|
| P1 | In pairs | 10 | Activity sheets | Kniwwelino board |
| P2 | Alone | 9 | Activity sheets + creativity material | Kniwwelino board + extensions |
| P3 | In pairs | 3–8 | Activity sheets | Kniwwelino board |
| S1 | Alone then in group | 11 | Developed its own education material | Kniwwelino board + extensions |
| S2 | Alone | 7 | Activity sheets | Kniwwelino board |
| S3 | NA | 0 | NA | NA |

*NA* Non-applicable

### 2.9.3 Pedagogical practices

According to the teachers, most of the pupils had no experience in programming, with just a few having already programmed on rare occasions (discovery via code. org in **P3**, discovery of Kniwwelino at an event in Luxembourg and a few lessons of Scratch with the pupils in **P2**). Four schools chose a progressive learning approach (**P1**, **P2**, **S1,** and **S2**). They first taught the basics of programming (matrix, LEDs, messages, variables, and loops). Then, they experimented with adding electronic extensions when possible (external LEDs, buzzer, servo motors). And finally, they called upon the children's creativity by giving them a challenge where they used their problem-solving skills. **P3** decided to train CT by leaving the children free in their discovery of Kniwwelino and programming by guiding them only in the way they investigated a topic and solved problems.

"The idea was that the pupils apply computational thinking skills, such as you do one step at a time, programming is an iterative process, and if you try to map out the whole program at once, it's gonna be really hard to find your mistake. So, if you put a block in, test it if it really works, and then you can add next block, and if it doesn't work it's really easy to fix your mistakes." (**P3**).

In **S3**, the plan was to use the KCK with secondary school pupils in the maker lab (outside of class time, in a space dedicated to electronics, computers, and do-it-yourself (DIY)) to design a model of a connected house using the extensions. The reasoning was that once trained, the high school pupils could teach programming to younger elementary pupils. However, this could not happen due to the lockdown.

### 2.9.4 Notes on SARS-CoV-2 (COVID sanitary crisis)

The study had to be interrupted due to the COVID crisis lockdown. Of the six schools participating in the study, three had finished their sessions before the lockdown, two had started sessions but were unable to finish, and one was unable to start, see Table 2. Those who continued (**S1-T6** and **S2-T8**) were high school teachers. They were able to provide each pupil with a Kniwwelino board so that they could continue working at home. Pupils were given instructions on how to connect their Kniwwelino board to their computer or tablet. Furthermore, they received support from teachers via Teams when needed. The interviews were conducted online with the teachers and the analysis of the questionnaires is based on those available in December 2020.

## 3 Study results

This section presents the results of the questionnaires and the thematic analysis of the interviews. The results are discussed according to the research questions specified earlier, and then according to the results of studies in the literature concerning tools to support CT teaching. See Fig. 5 for an illustration of the use of the KCK in classroom. Quotations have been given to illustrate the statements, however, not all quotations are shown. Quotations are given in the original language and translated.

## 3.1 Q1: In what way does the KCK support teachers' pedagogical practices?

To answer this question, we asked to teachers to identify strengths and weaknesses of the KCK regarding materials, the visual programming interface, and educational material.

### 3.1.1 Feedback on the materials

The materials are appropriate for classroom use. "*Le kit est bien tel qu'il est pensé*" (The kit is well thought out) (**P1**). The boxes are easy to carry and store. Several teachers mentioned that for younger pupils (late kindergarten, early elementary), the boards alone are sufficient and that using the extensions requires learning too many new things at once for this age group. Teachers pointed out that it was difficult to share a KCK between several classes at the same school because this requires a lot of coordination between teachers. Some teachers were faced with defective boards (**P1**, **P2**, **P3**). The boards could not restart properly because the pupils disconnected and reconnected them too quickly. The problem was solved by resetting the boards with the tool developed for this purpose. Two teachers pointed out problems with the wires: wires that broke too quickly when handled by children (**P2**) or pupils who found it too hard to use alligator clips since they were only familiar with USB connection (**S1**).

### 3.1.2 Feedback on the visual programming interface

The visual programming interface was found to be easy to use, clean, simple, and colorful. "*It's clean and really simple, I love the little tutorials that pop up.*" (**P3**) One teacher from **S1** noted that the visual programming interface allowed children to understand the logic of programming. The pupils feel like they are playing with a puzzle. "*Une introduction à la programmation avec un feedback en hardware, c'est une approche qui est très bien pour les élèves.*" (An introduction to programming with hardware feedback is a great approach for pupils.) (**S2**) However, teachers note that the lack of direct feedback in the visual programming interface is unusual for pupils. "*In code.org you have instant feedback (cheering and clapping). In*



**Fig. 5** Using the Kniwwelino Classroom Kit in pilot schools to teach computational thinking

*real programming you don't have that. The only feedback is whether or not it works. When it doesn't it can be frustrating. We had to make this clear in the sessions, and then we experienced real success.*" (**P3**).

### 3.1.3 Feedback on Wi-Fi usage

Many problems were caused by the Wi-Fi connection. These difficulties were mainly due to the technical infrastructure of the schools being very protected or the fact that the Wi-Fi signal was not strong enough in some classrooms. This resulted in download problems and frustration for the children. To address these issues, three schools developed their own dedicated Kniwwelino network, and two schools ended up borrowing the 4G router originally proposed. Teachers also reported that downloading the code to the boards via Wi-Fi was sometimes slow, causing frustration. "*Les deux premières séances d'une heure ont été utilisées à se connecter. Ça n'a pas marché pour tout le monde, le problème est lié à la connexion internet de l'école* via *[notre fournisseur].*" (The first two one-hour sessions were used to set up the connection. It didn't work for everyone; the problem is related to the school's internet connection via [our provider].) **P2**.

All participants agreed that the possibility to exchange messages between several boards thanks to Wi-Fi was a strength, adding fun and excitement for kids. "*Sending messages was insanely popular: What they preferred what sending messages to each other then, light matrix with their name of smiley face. It was the first thing they learned, and I did not even tell them the Kniwwelino could do it.*"(**P3**).

To provide increased autonomy, **S2** developed a QR-code-based connection system for the Kniwwelino board.

If **S3** had been able to develop a house IoT model, it would have been able to use the Node-RED possibilities to exchange messages with IoT items.

### 3.1.4 Feedback on educational material

The manual provided in the KCK was found to be well-adapted, useful, and clear. It allowed teachers to start their activities easily. The teachers mainly used the paper version. A few teachers adapted the activity sheets before printing them. In particular, some teachers supplemented the approach proposed in the KCK by adding debugging and code-checking exercises.

> "*Le petit classeur aide bien à démarrer.*" (The little handbook is a good way to get started.) (**P1**)
> "*Les supports que vous avez fournis étaient très aidant. On a utilisé le handbook et les tutoriels sur le wiki. On les a adaptés au fur et à mesure.*" (The materials you provided were very helpful. We used the handbook and the tutorials on the wiki. We adapted them as we went along.) (**S2**)
> "Exercise sheets are great and easy to copy." (**P3**)

At **P3**, teachers used code.org at first, then applied the knowledge gained to the KCK. The activity sheets were available for pupils to review. At **S1**, the teacher developed his own curriculum composed of 75 progressive lessons: 50 about programming and 25 about electronics.

### 3.2 Q2: How does the KCK fulfill the pedagogical objectives set by teachers?

To check whether the KCK enables the teachers to fulfil their pedagogical objectives, we collected the skills developed by pupils.

The pilot schools' use of the KCK identified, on average, pupil skill development in programming and electronics, problem solving and logic, creativity, collaboration, and empowerment, see Table 2. **P3-T4** was not able to make much progress due to the lockdown that ended the sessions prematurely and **S3-T9** was not able to begin. The following sections expand on the teachers' observations of the skills developed based on the interviews. In addition to the skills listed in the Table 2, P3 also noted an improvement in pupil tenacity. "*Perseverance. Sticking with a problem and trying to solve it.*" (**P3**)

#### 3.2.1 Learning programming and electronics

All teachers observed that pupils improved their programming and electronics skills, although they did not always meet their end goal. In particular, they observed that they integrated programming logic. "*Ils partaient de rien, et après il savaient manipuler, trouver les erreurs avec le feedback (…) Ils comprenaient la différence entre* "au démarrage" *et* "en répétition continue"*, c'est le plus important au départ.*" (They started from nothing, and then they knew how to use it, how to find errors with the feedback (…) They understood the difference between "*on start*" and "*repeat forever*" and that is the most important thing at the beginning.) (**S2**). Teachers also observed that pupils began to learn and then adopt the vocabulary of programming, as in (Grover, 2011): "*Once they had those words available to them and could see them, they started to use them a lot more.*" (**P3**).

#### 3.2.2 Learning problem solving

The pilot **P3**, which focused specifically on problem solving, saw significant growth in this skill as well as in the development of their perseverance and investigative skills. "*It was a huge enquiry for the pupils. An enquiry into how they could be problem solvers and what they could get Kniwwelino to do.*" (**P3**) The other pilot schools also observed that pupils were becoming increasingly independent in finding solutions to their problems. "*Ils ont eu des soucis, donc ils ont commencé un petit peu à chercher des solutions. Même si c'étaient des soucis techniques au départ, parfois ils cherchaient aussi leurs propres solutions pour ça, ça c'était pas inintéressant.*" (They had problems, so little by little, they started to look for solutions. Even though they were more technical issues at first, sometimes they were also looking for their own solutions for those, and it was quite interesting.) (**S1**)

### 3.2.3 Learning creativity

The teachers were able to observe creativity in exploring the different elements of Kniwwelino. They saw pupils try out different colors and display different symbols on the matrix. At **P1**, creativity could not be explored because of the lockdown. At (**P2**), the teacher pointed out that her pupils thought up overly complex projects that they were unable to develop, which frustrated them.

> "*On voyait qu'ils développaient leur esprit créatif. Il faut leur laisser l'espace pour. Dès qu'on ouvre un peu, la créativité commence.*" (We could see that they were developing their creative minds. You have to give them the space to do that. As soon as you open it up a little bit, the creativity starts). (**S2**).
> "Some of my kids got really into it. It showed them that they could be creative with a device like that." (**P3**).

### 3.2.4 Learning collaboration and social interactions

The pilot schools observed that with the use of the KCK, and working in pairs, collaboration and social interaction skills improved. This was also due to the ability to exchange messages through Kniwwelino matrices and the help that pupils spontaneously gave each other. "*Le partage a commencé surtout avec l'envoi de messages.*" (Sharing started first and foremost with messaging.) (**S2**)

### 3.2.5 Getting pupils interested in IT careers

According to the post-use questionnaire distributed to the pupils who participated in the pilot study, most of them want to: become developers (59.14%), develop other projects with Kniwwelino (67.39%), and try out other programming tools (85.87%), see Fig. 6. They find programming fun (87.23%).

### 3.3 Q3: What is the user experience of the KCK (for teachers and pupils)?

According to the feedback from teachers in interviews, teachers enjoyed using the KCK in their classrooms. Furthermore, they observed that the children also enjoyed the experience and had a lot of fun. They also strongly appreciated the tangible aspect of Kniwwelino "*C'est mieux que Scratch car on voit directement ce qu'on fait.*" (It's better than Scratch because you can see what you are doing directly.) (**S1**)

### 3.3.1 Teachers' user experience

Analysis of the UEQ results for the first version of the KCK prototype shows positive values for all scales: *Attractiveness* (average 1,596 with min -3 max +3 / standard deviation 0.71), *Perspicuity* (1,194 / 0.85), *Efficiency* (1,028 / 0.59), *Dependability* (1,222 / 0.76), *Stimulation* (2.0 / 0.66), and *Novelty* (1,667 / 0.81). We can conclude that the prototype gives a good first impression in terms of user experience. However, the *Efficiency* must be improved. The *Stimulation* scale has a very good score of 2. In Fig. 7, we
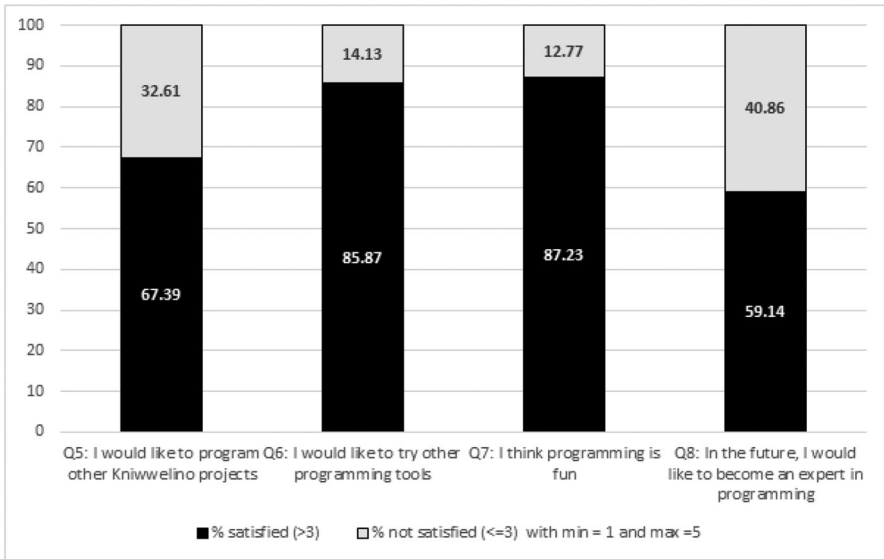
**Fig. 6** Pupils' interest in programming as a result of their use of the KCK in class

present the averages obtained (minimum -3, maximum 3) for each item of the UEQ. The items **Interesting** (2.4), **Valuable** (2.3), and **Good** (2.1) obtained the highest scores. The items **Easy** (0.7), **Predictable** (0.6), and **Fast** (0.2) obtained the lowest scores. However, we can note that for all items, the result always remains positive.

The UEQ also allows a new product to be compared in terms of user experience with a large sample of other widely used products. This benchmark database is divided into five categories: *Excellent*, *Good*, *Above average*, *Below average,* and *Bad* (Schrepp et al., 2014). Figure 7 shows the position of the KCK with respect to the items of this benchmark. The KCK is: **Excellent** concerning *stimulation* and *novelty*, **Above average** concerning *attractiveness* and *dependability*, and **Below average** concerning *perspicuity* and *efficiency*. None of the scales scored a *Bad* rating.

**Reuse and transferability** An important part of tool appropriation concerns whether its users want to continue using it and whether they think the tool can be transferred to others. All teachers involved in the study plan to use the KCK again next year (if they have the same classes). Furthermore, the teachers believed that the KCK can be used outside of programming instruction in subjects such as math, science, or even French. In addition, other teachers in the pilot schools have shown interest in using the KCK in the next school year.

"On a été jusqu'aux variables mais on a dû arrêter à cause du confinement. On va le réutiliser en 8O et prendre le volet électronique avec (...) pour des projets beaucoup plus individuels avec les sensors." (We went as far as the variables, but we had to stop because of the lockdown. We are going to reuse it in 8O and take the electronic part with (...) for much more individual projects with the sensors). (**S2**).
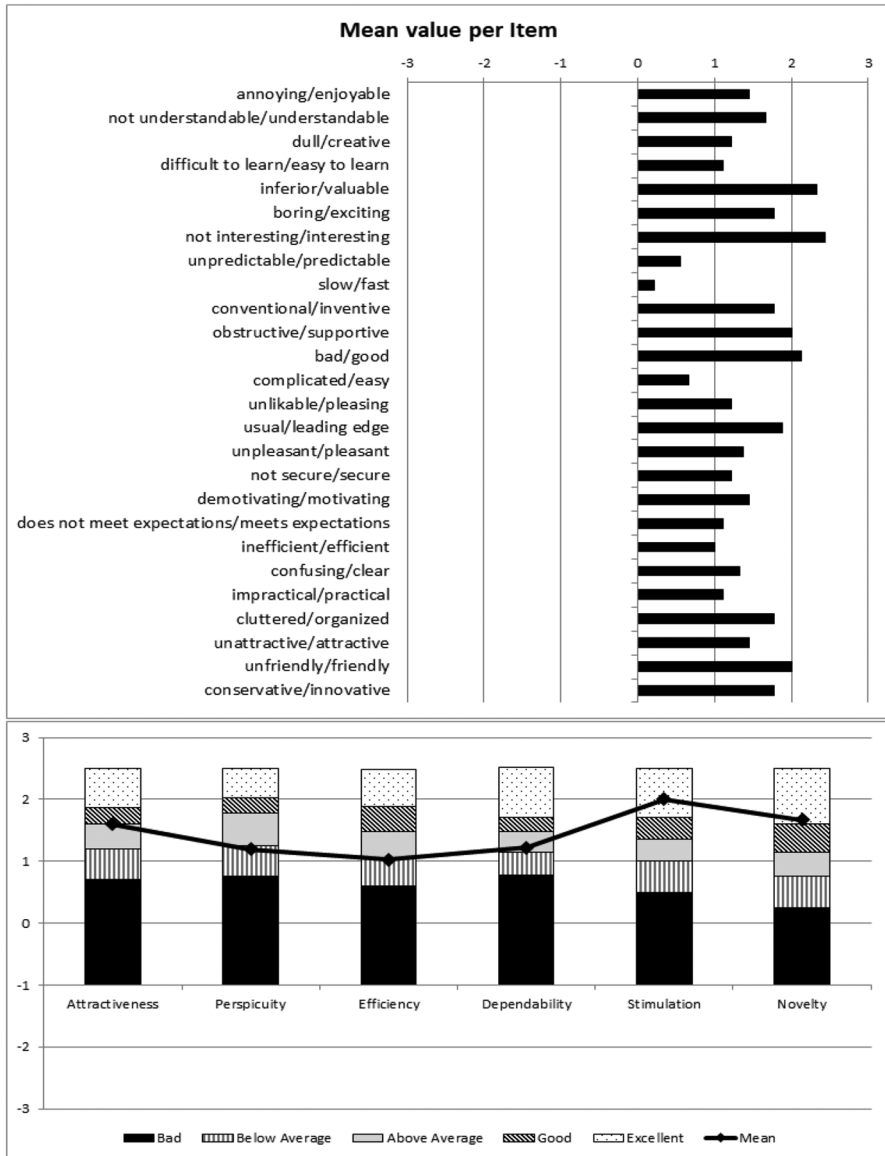
**Fig. 7** On the top: Results of the teachers' responses to the UEQ concerning the first version of the KCK prototype, by item.—On the bottom: comparison of the KCK with the benchmark provided by UEQ

"Il y a des petites passerelles : ça apprend la logique, ça peut avoir un lien avec les maths, les connecteurs logiques, ça peut avoir un lien avec les langues, le Français." (There are small bridges: it teaches logic, it can have a link with math, logical connectors, it can have a link with languages, French). (**P1**).

Regarding KCK transferability, teachers were very satisfied with the initial Kniwwelino training and believed that it would be useful for teachers using the KCK in the future.

However, they also pointed out the complexity of the initial implementation of Kniwwelino. The teachers believe that success depends on the teacher's experience, attitude, and mindset in solving technology problems. They recommended that a group of teachers act as a resource for other teachers in each school. The experience of using the KCK was so enjoyable for **P1** that they launched a project with four schools in Europe to explore and promote the use of Kniwwelino, especially through the use of Information and Communication Technologies in the teaching of language, mathematics, science, and art for primary level pupils (9–10 years old).

### 3.3.2 Pupils' user experience

94 pupils responded to the post-use questionnaire of the KCK in class. The results show that the pupils were satisfied with their use: 82.98% liked programming with Kniwwelino, 86.02% found programming with Kniwwelino exciting, 81.72% thought it was easy to program with Kniwwelino, and 76.6% thought that anyone could program with Kniwwelino, see Fig. 8.

Our observations on pupil experience show that the overall user experience was positive for both pupils and teachers, and the UEQ results are quite encouraging for a first prototype.
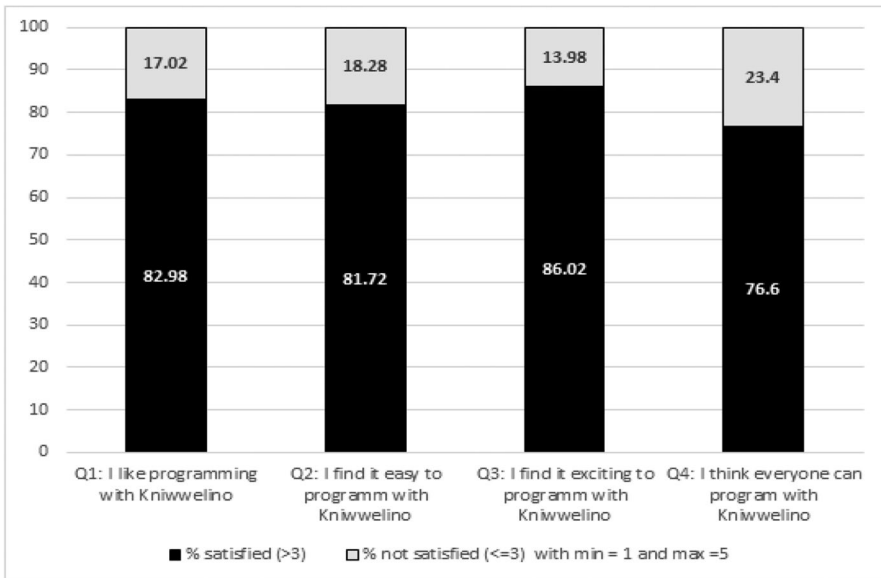


**Fig. 8** Pupils' satisfaction with their experience using the KCK in the classroom

**Pupils' motivation** Two schools (**S2** and **P1**) mentioned that the pupils were not very motivated at the beginning but became more interested as they progressed. **P1** justified this by the fact that progress was very slow at first, but once the technical problems had been mastered, the pupils had more time to program. In addition, once the pupils were able to master several elements of Kniwwelino, they had fun and played with it. The same was true for **P2**, where there were many technical problems at the beginning that negatively impacted the experience of the pupils who were very motivated, however, after the technical problems had been solved, the pupils had fun.

"Il y a des élèves qui ont commencé avec beaucoup de peur, mais dès qu'ils ont fait le premier pas, ils y ont pris goût, surtout les filles." (There are pupils who started with a lot of fear, but as soon as they took the first step, they got a taste for it, especially the girls). (**S2**).

"Au début c'est vrai qu'il y avait pas une grosse grosse motivation, sûrement en rapport un peu avec ça (les problèmes de connexion au réseau de l'école) aussi. Puis au fur à mesure qu'ils ont commencé à mieux maîtriser, pour pouvoir explorer un peu plus eux-mêmes, là ils étaient plus motivés, ça bougeait plus." (At the beginning, it's true that there wasn't a lot of motivation, probably also related to that (the problems of connecting to the school network). Then, as they started to master it better, to be able to explore a little more themselves, they were more motivated, it became more dynamic). (**P1**).

"Au départ aussi c'était assez limité, quand on fait les premières séances, où il y a qu'un type de programme à faire, vous travaillez sur un élément, bon voilà, ils ont vite fait le tour. Mais quand on commence à intégrer plus d'éléments dedans, du coup ils peuvent plus jouer et créer des choses et du coup cela les intéresse plus." (At the beginning it was also quite limited, when we do the first sessions, where there is only one type of program to do, you work on one element, and there you go, they quickly did everything. But when we start incorporating more elements into it, suddenly they can play more and create things and suddenly they're more interested.) (**P1**)

For **P3**, at first the pupils were very frustrated because they lacked direct feedback in the visual programming interface: they expected to find out in the interface directly whether their program was correct or not, as is the case in code.org or Scratch. Then, they started to have fun, especially when they discovered the message exchange.

According to the teachers' observations, the pupils particularly appreciated the fun approach, which strongly motivated them, as well as the fact that they could learn progressively. "*Les élèves aiment le côté ludique, l'approche par le jeu. Ils aiment la progression,* "on avance, c'est cool". *L'apprentissage est facile et logique.*" (The pupils like the fun aspect, the game approach. They like the progression, "*we're moving forward, it's cool*". Learning is easy and logical.) (**S1**) pupils' favorite activities were changing the color of the LEDs, displaying icons on the LED matrix as emoticons, creating animations, and exchanging messages.

# 4 Discussion

In this section, the observations made will be discussed and we will resume them as recommendations. Then, our results will be discussed regarding the studies listed in the state of the art on tools used for teaching CT.

## 4.1 Recommendations and lessons learned

From the observations of the study, we can draw up recommendations and plans for future improvements of the KCK.

As far as our first research question is concerned, that is **the way in which the KCK supports the pedagogical practices of teachers**, our observations of the design of the KCK are related to the materials comprising the KCK, the associated visual programming interface, the usage of Wi-Fi, and the educational material.

We observed that, as the material is not stored in classrooms, it needs to be stored in a way in which it is easy to carry it from its storage place to the classrooms. The **KCK design**, with all pupils' boxes in a big storage container, is particularly well-adapted for this. If pupils use Kniwwelino in the classroom only during school time, it is better to have one KCK for two classrooms maximum, and ideally one per classroom.

Regarding the **contents of the pupil boxes**, we noticed that, in primary schools, with young pupils (4 to 7 years old), only the tangible board was used, and not the electronic extensions. Deducing that the tangible board is sufficient for the context of the first years of primary school and that electronic extensions are thus not necessary, we could propose a "*lighter version*", targeting younger pupils, which would contain only two extensions or no extension at all. For older pupils and higher classes that do use electronic extensions, we should provide easier and more robust ways to connect the extensions to the board.

The users of our pilot all appreciated the puzzle as a **visual programming interface** (as with Scratch). Teachers considered it particularly suitable for teaching CT. We therefore recommend a clean, simple, and colorful visual programming interface. A tool exists to reset the code from the boards. It should be made available to the teachers and should be easy to use.

The opportunity to give feedback on the code (code is true or false) was requested by the pupils, while the teachers praised the fact that the code needed to be uploaded to the tangible board to be checked. Adding this feature to the programming interface remains an open question and requires further experimentation.

The **Wi-Fi connection** enables pupils to exchange messages between boards and even with IoT items, a feature they particularly appreciated and that contributed to a great user experience. However, Wi-Fi infrastructures are not well developed in schools in Luxembourg and setting the Wi-Fi connection proved complicated. This made a bad first impression on pupils and hindered their overall experience. Kalogiannakis also recommends a reliable connection to avoid issues when an online compiler is used (Kalogiannakis et al., 2021). To overcome this difficulty, we recommend providing a Wi-Fi router as an option, in order to facilitate the Wi-Fi connection. Some teachers at higher educational levels removed this difficulty by

teaching their pupils what Wi-Fi was and how to use it. Integrating this dimension to the existing curriculum, as well as the exchange of messages and MQTT, would enrich learning and should be further investigated.

The **educational material** provided in the KCK, namely the pedagogical booklet, consisted of progressive lessons in coding and electronics, and is available in a paper and adaptable and downloadable format. This booklet proved a valuable tool for all teachers, who either copied the paper version, or printed the online version to give to their pupils. Granting access to a curriculum available both online and as a printed or at least printable resource is a good practice. Some teachers even adapted the lessons in the booklet to their own learning objectives and practices. Thus, to improve the adaptability of the booklet, teachers should be given the editing rights to modify, save and print lesson sheets, and even to share them with their colleagues. It was noted by Kaliogiannakis that most teachers first used the official activities, even if some of them developed their own additional pedagogical material (Kalogiannakis et al., 2021). The authors argue that this is certainly due to a lack of confidence on the part of the teachers. In our study, we saw also that the teachers more confident in CT developed their own curriculum (for **S1**) or a new approach (enquiring for **P3**).

The **COVID context** of our study highlighted that the KCK was used both in a classroom and home setting (for homework and during lockdown). We should develop a fully online or blended curriculum, mixing classroom exercises and work at home, in order to enrich the pupils' use experience and support more learning practices. The fact that the programming interface can be opened in a browser with no need to install software and that it can be used on a computer, tablet, or smartphone, are crucial advantages.

As far as our second research question is concerned, that is, **how the KCK helps teachers fulfill the learning objectives they set**, our observations show that the KCK fulfilled all CT-related learning objectives set by teachers, including debugging, programming, and electronics. Practicing CT leads to the adoption of the specific CT language. The KCK, used in classrooms, also supports the learning of soft skills, such as creativity, collaboration, communication, social interactions, and empowerment, as can be seen in Fig. 9. Finally, the KCK was also used to raise awareness of IT careers.

For our third research question, our observations show that the **user experience of the KCK** is positive for both pupils and teachers, and the UEQ results are encouraging for a first prototype. According to the feedback from teachers in interviews, teachers enjoyed using the KCK in their classrooms. Furthermore, they observed that the children also enjoyed the experience and had fun. They also strongly appreciated the tangible aspect of Kniwwelino "*C'est mieux que Scratch car on voit directement ce qu'on fait*." (It's better than Scratch, as you can see what you are doing directly) (**S1**).

Regarding **the user experience of teachers**, they consider the KCK very flexible, as it can be adapted to diverse ways of organizing learning sessions, offering a wide range of uses to teachers:

- It can be used by pupils individually, in pairs, and in groups;
- Teachers usually use it for 8–11 sessions, focusing on the main board and using the activity sheets;
- It can be used in a classroom setting, for homework, and for distance learning (teachers continued to use it during lockdown).
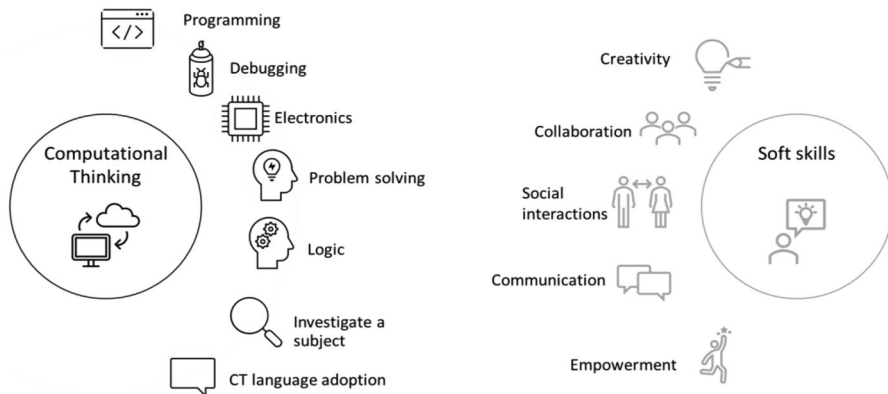
**Fig. 9** Pedagogical objectives observed to teach CT in class with Kniwwelino

Although the **initial KCK training** was not necessary for all teachers, it was highly appreciated by all those who participated in it and should be part of the final setting. This training should be optional, free of charge, and provided by the official teacher training body. It would ensure that all teachers have the same level of knowledge in using the KCK, and would also enable teachers to network, discuss difficulties, and benefit from expert users' feedback and experience. As a consequence, a community of teachers should be built up to observe and report on best practices and to act as a support, maybe limited to one school or to one educational level within a region. It would create bonds among the community, thus fostering its sustainability.

Our observations show that **pupils' motivation** increased after just a few sessions. We suggest building curricula of more than one session to motivate pupils to learn CT rather than a one-off session. Regarding motivation, Kaliogiannakis et al. noted that it was high at the beginning but may decrease over time (Kalogiannakis et al., 2021). However, in our study, it was the contrary. They also noted that highly capable students are less enthusiastic due to the device's limited capabilities. This can maybe explain the disappointment of pupils from **P2** during the creativity phase, where they designed projects that were too complicated to be implemented with Kniwwelino.

All recommendations issued from these observations are summarized in Fig. 10.

## 4.2 Discussion of the study results in relation to other support tools for teaching computer thinking

As in our study, Kaučič's study of Scratch involves primary and secondary school pupils (Kaučič & Asič, 2011). To answer Kaučič's questionnaire, pupils rated the items on a 5-point Likert scale ranging from 1 (total disagreement with the statement) to 5 (total agreement with the statement). If we try to compare our results with Kaučič's results for Scratch (see Table 4), we can observe that pupils enjoy programming with Kniwwelino (3.90 vs. 3.81 for Scratch). Scratch seems easier to
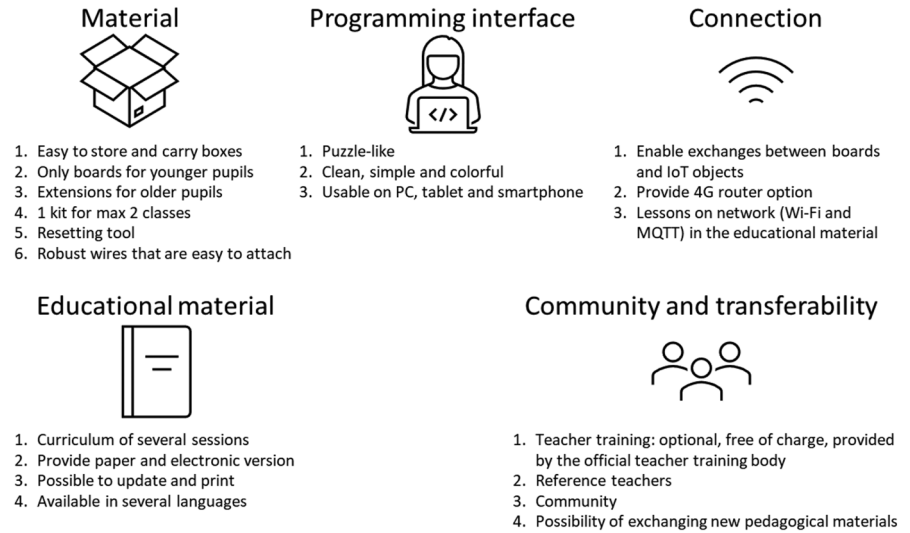
**Fig. 10** Recommendations for TDP development to teach CT in class

**Table 4** Comparison between Kaučič's study with Scratch (Kaučič & Asič, 2011) and our study with the KCK

| Kaučič's study with Scratch item | Score | Our study with KCK Item (and translation) | Score |
|---|---|---|---|
| Scratch is attractive (I like it) | 3.81 | J'aime bien programmer avec Kniwwelino<br>*I like programming with Kniwwelino* | 3.90 |
| Scratch is easy to use | 3.94 | Je trouve ça facile de programmer avec Kniwwelino<br>*I found easy to program with Kniwwelino* | 3.62 |
| Scratch was fun to work with | 4.03 | Je trouve ça excitant de programmer avec Kniwwelino<br>*I found it exciting to program with Kniwwelino* | 3.90 |

use for programming (3.94 vs. 3.62 for Kniwwelino) and slightly more fun (4.03 vs. 3.90 for Kniwwelino). However, the comparison should be interpreted with caution: the questions are not exactly the same, the number of participants is not the same, and even if the contexts seem similar, they are not the same. Nevertheless, it shows that Kniwwelino provided a similar experience for pupils.

In Shina's study on discovering programming with Scratch, 60% of pupils surveyed found Scratch easy to use (Schina et al., 2019). In comparison, in this study with Kniwwelino, the percentage of satisfied pupils that found Kniwwelino easy to use is 80.65% (see Fig. 8).

In Gibson's study, BBC Micro:bit was evaluated by pupils in two elementary schools over one or two days of use (Gibson & Bradley, 2017). Pupils rated the enjoyment of using BBC Micro:bit on a Likert scale ranging from 1 (very annoying) to 5 (very enjoyable). 92% of pupils in School A and 90% of pupils in School B reported that using BBC Micro:bit was either enjoyable or very enjoyable. The BBC Micro:bit

results are somewhat better than those obtained for the question "*I find it exciting to program with Kniwwelino*" in our study, where 86% were excited or very excited (see Fig. 8). However, as mentioned above, the results cannot be compared exactly.

### 4.3 Limitations and problems encountered

There are several aspects of this study that need to be mentioned because they might limit the utility of our findings. Concerning the questionnaires we used, they do not allow us to compare the difference per child before and after the study, or to correlate with other measurements. Since the data collection was conducted by the teachers, we cannot be certain which information was provided at what moment and there might be differences between the participating schools and classes. In addition, not all questionnaires were completed and returned.

Regarding the participants, only motivated and engaged teachers were recruited and results might be different with other teachers. However, it can be noted that the students were not selected based on their motivation, but only because they were part of the class in which the teacher worked.

Finally, the study was interrupted due to the COVID-19 pandemic which imposed on teachers a radical change in their teaching methods with a switch to home schooling. Due to this interruption, the program could not proceed as initially planned.

## 5 Conclusion

Teaching computational thinking (CT) is strategically important for developing certain skills needed in the future and different tools are available to support CT learning, including powerful Tangible Development Platforms (TDP). However, few studies report on the use of TDP for learning CT and to our knowledge, none of them refers to the use of this type of tool for teaching CT (at least, from the teachers' point of view). Therefore, since the teaching of CT was being integrated into the official curriculum in primary and secondary schools in Luxembourg, we used the opportunity to conduct a pilot study to observe the relevance of using a TDP to teach CT in the classroom. We adapted Kniwwelino to create the Kniwwelino Classroom Kit (KCK), which is a TDP designed for children including an electronic board and electronic extensions, a visual programming interface, and pedagogical and creative materials developed in Luxembourg. The KCK was tested over one school year (2019–2020) by six schools (three primary schools and three secondary schools throughout Luxembourg), represented by eleven teachers and used by 138 pupils. The purpose of this pilot study was to answer three research questions: (1) in what way does the KCK support teachers' pedagogical practices, (2) how does the KCK fulfill the pedagogical objectives set by teachers and (3) what is the user experience of the KCK for teachers and pupils. To answer these research questions, we used a mixed method approach including qualitative methods (logbooks and interviews of teachers) and quantitative methods (questionnaires: pre and post usage questionnaires for pupils, UEQ questionnaires for teachers). After introductory training on

the KCK for the volunteer teachers, we left them free to organize the programming sessions at their convenience with a minimum of three sessions and let them freely adapt the pedagogical material and curriculum included in the KCK. Although COVID and the lockdowns somewhat disrupted the experiment, we were able to answer the research questions.

(1) As observed, pedagogical practices were varied in terms of the number of sessions, whether or not electronic extensions were used, whether teachers used the proposed curriculum or developed their own, and whether pupils worked in pairs or alone. It was used on school premises and for homework, on different types of devices (computer, tablet, phone) without installing software. Based on this study, the KCK support the teachers' pedagogical practices in primary and secondary schools in Luxembourg.

(2) The pedagogical objectives pursued by teachers were also varied, including CT, programming, electronics, problem solving, and logic, as well as soft skills such as as creativity, collaboration, social interactions, communication, and empowerment, and finally, stimulating interest in IT careers. According to the assessment done by teachers, their objectives were mainly fulfilled. Indeed, some were surprised to see that pupils also acquired knowledge on topics that were not initial pedagogical objectives, such as P2-T3 about programming. When objectives were not entirely fulfilled, it was mainly due to the COVID and the lockdowns which disrupted the school rhythm. We also note that when this study took place, CT had not yet been officially included in the national curriculum. This may explain why teachers preferred to concentrate on other subjects during these troubled times.

(3) Regarding the UEQ results from the teachers, the KCK provides a good user experience particularly concerning its stimulation, novelty, and attractiveness. KCK was evaluated as interesting, valuable, and good. But some improvements should be carried out to make it easier, more predictable, and faster. Teachers also reported that they using the KCK outside of programming instruction could be interesting for subjects like math, science or French. Teachers believe that the KCK is a perfect introduction to programming and would recommend it to their colleagues to give their pupils a first contact with computational thinking and logic, before using programming languages. As for the pupils, they liked the KCK, and found programming with it easy and exciting and thought everyone would be able to program with it. Although the start was slow and sometimes complicated, pupils appreciated the fun approach and found it highly motivating, particularly the possibility to exchange messages between two boards. These observations concerning the pupils' user experience are consistent with other studies that use a TDP like BBC Micro:bit or visual programming like Scratch to learn CT in a school context in terms of enjoyment, ease of use, fun, and attractiveness (Kaučič & Asič, 2011, Schina et al., 2019, Gibson & Bradley, 2017). Nevertheless, the comparison has to be interpreted with caution as the latter studies are different from the one described in this paper in terms of number of participants, context, and questions asked to pupils.

The main limitations and problems encountered during this pilot study concerned the limited control over the study procedure conducted by the teachers, as well as the interruption of the study due to COVID-19. Nevertheless, our work describes a concrete use case and shows how the teachers and pupils reacted in the face of real challenges that occurred.

Based on the results of the study and observations made, we propose the following recommendations for using a TDP to teach CT in a classroom setting based on five dimensions.

(1) the material must be designed so that it is easy to store and to carry. The boxes should contain only boards for younger pupils and boards and some electronic extensions for the older pupils and easy and robust wires to plug into the boards. One kit can be shared by up to two classes. A resetting tool should be proposed.
(2) the programming interface must be visual and puzzle-like, clean, simple, and colorful, as well as usable on a PC, tablet, or smartphone.
(3) the connection must enable exchanges between boards and IoT objects, a 4G router option should be provided, and lessons on network (Wi-Fi and MQTT) should be included in the educational material.
(4) the educational material must contain a curriculum consisting of several sessions, be provided in paper and electronic versions, be adaptable (if necessary) and printable, and be available in several languages (particularly important in the Luxembourgish context).
(5) and the community and transferability are improved by a training session for teachers that must be optional, free of charge, and provided by the official teacher training body. A community must be built, including reference teachers and the possibility of exchanging new teaching materials.

This pilot study shows that the strength of the KCK is its versatility. The KCK supports any learning practices, and can be adapted to any circumstances, even during the COVID crisis. We should therefore develop a fully online curricula to support online and blended learning.

The impacts on pupils are multiple, with an increase in knowledge and skills in programming, problem solving, creativity, and collaboration. However, we must also ask ourselves whether it was rather the discovery of a new activity (i.e., programming) that sparked pupils' interest. Another impact, which was completely unexpected, is the reduction of the gender gap in programming. Indeed, some teachers pointed out that the KCK was attractive to both girls and boys. "*Dès que ça devient créatif, on a les filles avec.*" (As soon as it gets creative, we get the girls on board.) (**S2**). "[…] *ils y ont pris goût, surtout les filles*." ([…] they got a taste for it, especially the girls.) (**S2**). This comment can be related to Videnovik et al., 2018 who observed an increase in attention and skill development in girls after using BBC Micro:bit and an increase in their enthusiasm for coding. But the author points out that this has certainly more to do with the opportunity offered by the course to discover and learn programming than with the tool itself.

Thus, in answer to our research question, our results show that the KCK is suitable for teaching CT in classes. Nevertheless, improvements must be considered, such as ease of installation and set-up, time to load code onto the board, and robustness.

The strengths of the KCK are the tangible character of the learning platform, the possibilities offered to increase creativity, notably thanks to the matrix, the LEDs and the exchange of messages, and the social nature of the tool thanks to working in pairs and the exchange of messages.

Finally, we should support the development of a community of teachers that use the KCK by giving them the opportunity and the dedicated space and time to discuss their practices and contribute to the KCK by sharing the curricula and materials they develop with their colleagues. The initial KCK training was the first step in building this community.

**Data availability** Data from the study are not available.

After consultation with our ethics committee, we decided not to publish any data. Indeed, the Luxembourg field is quite small, and the participants could be easily identified.

## Declarations

**Competing interests** We have no conflicts of interest to disclose.

## References

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A k-6 computational thinking curriculum framework: Implications for teacher knowledge. *Journal of Educational Technology & Society*, 19(3), 47–57. http://www.jstor.org/stable/jeductechsoci.19.3.47

Arduino. (2022). Arduino website. https://www.arduino.cc/. Accessed 03 July 2023

BBC. (2022). BBC Micro:bit website. http://microbit.org/. Accessed 03 July 2023

Bdeir, A. (2009). Electronics as material: Littlebits. *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*, 397–400. https://doi.org/10.1145/1517664.1517743

Boyatzis, R. E. (1998). Transforming qualitative information: Thematic analysis and code development. *Sage*.

Calliope. (2022). Calliope website. https://calliope.cc/. Accessed 03 July 2023

Ching, Y.-H., Hsu, Y.-C., & Baldwin, S. (2018). Developing computational thinking with educational technologies for young learners. *TechTrends, 62*(6), 563–573. https://doi.org/10.1007/s11528-018-0292-7

Foundation, O., & Contributors. (2022). Node-red website. https://nodered.org. Accessed 03 July 2023

Gibson, S., & Bradley, P. (2017). A study of Northern Ireland key stage 2 pupils' perceptions of using the bbc micro: Bit in stem education. *The STeP Journal*, 4(1), 15–41. https://ojs.cumbria.ac.uk/index.php/step/article/view/374/495

Grover, S. (2011). *Robotics and engineering for middle and high school students to develop computational thinking*. Annual meeting of the American educational research association.https://doi.org/10.3916/C59-2019-06

Hall, L., Hume, C., & Tazzyman, S. (2016). Five degrees of happiness: Effective smiley face likert scales for evaluating with children. *Proceedings of the The 15th International Conference on Interaction Design and Children*, 311–321. https://doi.org/10.1145/2930674.2930719

Hsu, T.-C., Chang, S.-C., & Hung, Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education, 126*, 296–310. https://doi.org/10.1016/j.compedu.2018.07.004

Johnson, R., Shum, V., Rogers, Y., & Marquardt, N. (2016). Make or shake: An empirical study of the value of making in learning about computing technology. *Proceedings of the The 15th International Conference on Interaction Design and Children*, 440–451. https://doi.org/10.1145/2930674.2930691

Kalogiannakis, M., Tzagkaraki, E., & Papadakis, S. (2021). A systematic review of the use of bbc micro: Bit in primary school. *Proceedings of the 10th Virtual Edition of the International Conference New Perspectives in Science Education,* 379–384. https://doi.org/10.26352/F318_2384-9509

Kastner-Hauler, O., Tengler, K., Sabitzer, B., & Lavicza, Z. (2022). Combined effects of block-based programming and physical computing on primary students' computational thinking skills. *Frontiers in Psychology, 13.* https://doi.org/10.3389/fpsyg.2022.875382

Katterfeldt, E.-S., Cukurova, M., Spikol, D., & Cuartielles, D. (2018). Physical computing with plug-and-play toolkits: Key recommendations for collaborative learning implementations. *International Journal of Child-Computer Interaction, 17*, 72–82. https://doi.org/10.1016/j.ijcci.2018.03.002

Kaučič, B., & Asič, T. (2011). Improving introductory programming with scratch? 2011 *Proceedings of the 34th International Convention MIPRO*, 1095–1100. https://ieeexplore.ieee.org/abstract/document/5967218

Laugwitz, B., Held, T., & Schrepp, M. (2008). *HCI and Usability for Education and Work: 4th Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society, USAB 2008, Graz, Austria. Proceedings 4* (pp. 63–76). Springer Berlin Heidelberg.

Lego. (2022). Lego website. https://www.lego.com/en-us/themes/mindstorms. Accessed 03 July 2023

LIST. (2022). Kniwwelino website. https://www.kniwwelino.lu/en/. Accessed 03 July 2023

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for k-12? *Computers in Human Behavior, 41*, 51–61. https://doi.org/10.1016/j.chb.2014.09.012

Maquil, V., Moll, C., Schwartz, L., & Hermen, J. (2018). Kniwwelino: A lightweight and wifi enabled prototyping platform for children. *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction* 94–100. https://doi.org/10.1145/3173225.3173275

MEN. (2021). *Lettre circulaire de printemps 2021. Circulaire ministérielle aux administrations communales concernant l'organisation de l'enseignement fondamental pour la rentrée 2021/2022*. Ministère de l'Éducation nationale, de l'Enfance et de la Jeunesse. ISBN: 978–99959–1–297–0.

MQTT. (2022). Mqtt website. https://mqtt.org/. Accessed 03 July 2023

Sáez-López, J.-M., Román-González, M., & Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using "scratch" in five schools. *Computers & Education, 97*, 129–141. https://doi.org/10.1016/j.compedu.2016.03.003

Scherer, R., Siddiq, F., & Viveros, B. S. (2020). A meta-analysis of teaching and learning computer programming: Effective instructional approaches and conditions. *Computers in Human Behavior*, 106349. https://doi.org/10.1016/j.chb.2020.106349

Schina, D., González, V. E., & Rodríéguez, M. U. (2019). Gender differences in students' feedback and performance in scratch programming. *Conference Proceedings EDUNOVATIC 2018: 3rd Virtual International Conference on Education*, 36. ISBN 978–94–92805–08–9

Schrepp, M., Hinderks, A., & Thomaschewski, J. (2014). Applying the user experience questionnaire (UEQ) in different evaluation scenarios. *International Conference of Design, User Experience, and Usability*, 383–392. https://doi.org/10.1007/978-3-319-07668-3_37

Sentance, S., Waite, J., Hodges, S., MacLeod, E., & Yeomans, L. (2017). " creating cool stuff" pupils' experience of the bbc micro: Bit. *Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education*, 531–536. https://doi.org/10.1145/3017680.3017749

Videnovik, M., Zdravevski, E., Lameski, P., & Trajkovik, V. (2018). The bbc micro: Bit in the international classroom: Learning experiences and first impressions. 2018 17th *International Conference on*

*Information Technology Based Higher Education and Train* 1–5. https://doi.org/10.1109/ITHET.2018.8424786

Wing, J. (2017). Computational thinking's influence on research and education for all. *Italian Journal of Educational Technology*, 25(2), 7–14. https://doi.org/10.17471/2499-4324/922

## Authors and Affiliations

**Lou Schwartz[1]** · **Valérie Maquil[1]** · **Laurence Johannsen[1]** · **Christian Moll[1]** · **Johannes Hermen[1]**

✉ Lou Schwartz
　lou.schwartz@list.lu

[1]　ITIS Department, Luxembourg Institute of Science and Technology (LIST), 5, Avenue Des Hauts-Fourneaux, L-4362 Esch-Sur-Alzette, Luxembourg