Guidelines for Introducing Learners to Computer Programming in a Developing Country

Jean H. Greyling(✉)

Department of Computing Sciences, Nelson Mandela University, Port Elizabeth, South Africa
Jean.Greyling@Mandela.ac.za

Abstract. The South African government has committed to the implementation of coding and robotics teaching in primary schools. This vision faces the challenge that 16,000 schools in the country do not have computer laboratories, and that most teachers are not trained to teach coding. To address this reality, the TANKS mobile app was developed at Nelson Mandela University in South Africa. It introduces introductory coding concepts without the need for a computer. The scope of the project has broadened since its initiation in 2017. Originally learners were introduced to the game, and each received a game pack. Currently coding kits with various additional resources are made available to schools, mostly after teachers are trained. Based on the evolvement of the project, generic guidelines for the introduction of computer programming in schools, are provided as the main contribution. These guidelines identify the tools to be used here, how to make the project financially sustainable, alternative methods to traditional teaching, as well as the role of training for this kind of coding instruction.

Keywords: Unplugged coding · Children coding · Coding tools · Disadvantaged schools

1 Introduction

The challenge of introducing learners to computer programming in South Africa is hampered by the harsh reality of its school landscape, where a reported 16,000 schools do not have computer laboratories [7, 32]. Furthermore, most of these schools are classified as "non-fee paying," which implies that they do not have the available budgets to spend on the installation of computer laboratories and the purchase of expensive equipment. It is estimated that, on average, it will cost one million rand per school to supply the 16,000 schools with internet-connected laboratories [7].

R. J. Barnett et al. (Eds.): SACLA 2022, CCIS 1664, pp. 75–90, 2022.
https://doi.org/10.1007/978-3-031-21076-1_5

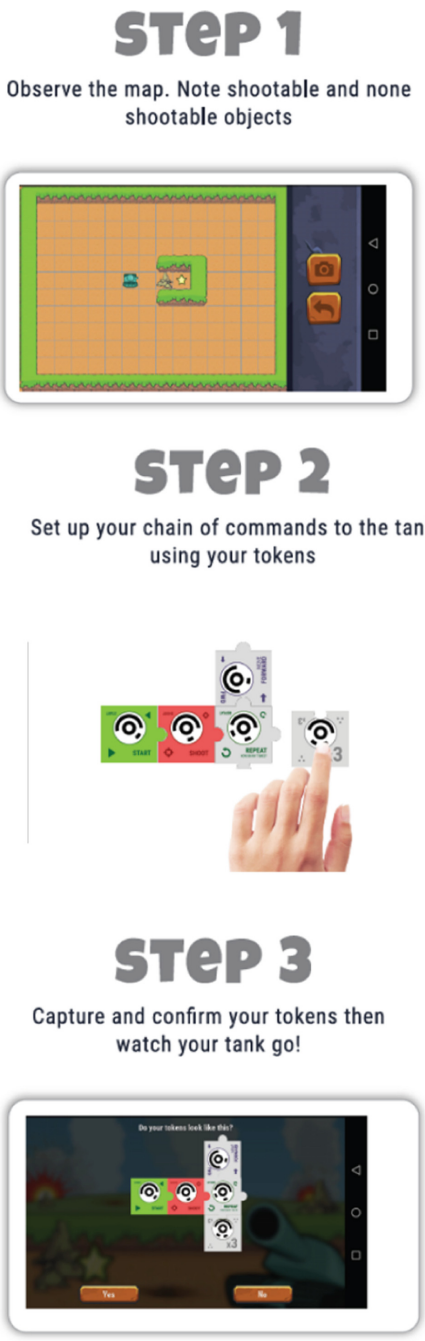

Fig. 1. TANKS app

Robots that are often suggested as suitable for teaching coding can cost up to R7,000 each, with a maximum of only five to six learners effectively being able to interact with such a robot at one time. In addition to technological challenges, the staff members' skills at schools are a further challenge. Few teachers have adequate coding training to properly introduce learners to conventional coding languages such as HTML, Python, and Java. Although the Department of Basic Education has many training initiatives, it is argued that these are insufficient for training teachers to have the desired educational impact. A further challenge is the availability of staff with enough technical knowledge to maintain computer laboratories. The poorly managed, improperly functioning laboratories found at schools across South Africa are a stark reminder of this reality.

Considering the ongoing Fourth Industrial Revolution and the shortage of relevant skills in South Africa, its government is actively driving the introduction of coding and robotics in schools from Grade R onwards. An investigation of the published draft curricula shows a heavy reliance on the availability of computers, other technologies (e.g., robots, circuits), as well as teachers with a solid understanding of topics such as coding and electronics. While South Africa is generally described as a developing country, many of its citizens do in fact have access to the infrastructure of a modern developed country. This creates a significant challenge to the country: the danger of continuously widening the divide between the "haves" and the "have-nots." This digital divide will be addressed within this article. Any coding and robotics curricula implemented in South African schools that fail to take into consideration the above-mentioned realities will be in

danger of actively widening the digital divide found among schools and learners in the country.

This article aims to find answers to the challenge of introducing learners to computer programming in a developing country by reporting on the rollout of the TANKS tangible coding app (Sect. 2), and how it evolved into a broader unplugged coding project (Sect. 3). Insights obtained from the four stages involved contribute to the provision of general guidelines (Sect. 4). Section 5 provides conclusions and future research.

2 The TANKS Coding App

The TANKS coding app was developed as an honors project at the department of computing sciences at Nelson Mandela University [2]. As shown in Fig. 1, it makes use of physical "tokens" that learners' piece together to build code without a computer. Mobile devices are needed (ideally 8 per class of 40), which are considerably less expensive than a 40 PC laboratory.

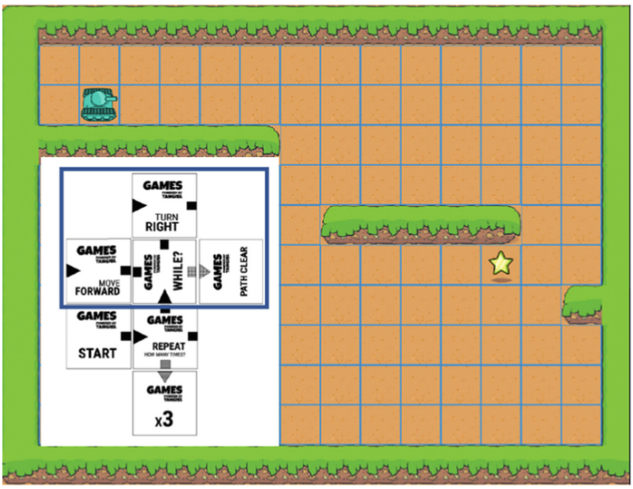

Fig. 2. Level 15 with solution

The user is given a challenge on the screen of a mobile device where a tank needs to move to a destination with obstacles in its way. The physical tokens are then used to compile the instructions to the tank, after which a photo is taken of the tokens. Using image recognition, the instructions are internalized, and then executed by moving the tank. There are 35 levels of increasing complexity. As learners progress through the levels, different coding concepts normally taught in an introductory programming module are introduced.

Various physical tokens were designed representing essential actions, including basic coding concepts. The basic tokens are used to move the tank forward and backwards and turn the tank to the left and the right. Furthermore, there are tokens that support

looping (repeat) constructs such as the "for" loop and the "while" loop, as well as a conditional coding construct ("if" statement). Complexity of the challenges is increased as looping and "if" constructs are introduced, followed by nested constructs. See Fig. 2 showing Level 15 which uses nested constructs – a "while" loop within a "repeat" loop. Identifiable markers (like QR codes) are placed on each token to allow for image recognition. The game is preferably played in workshops where learners participate in groups. Its gamification concept creates an exciting, fun atmosphere. Section 3.1 provides the theoretical background behind the design of TANKS and is part of a discussion of the four stages involved in rolling out a broader unplugged coding project.

3 Stages in Rolling out the Unplugged Coding Project

With the ultimate goal to provide generic guidelines (Sect. 4), the development of the unplugged coding project from the initial implementation of the TANKS mobile app, its rollout, the extension of tools, and teacher training are discussed in this section.

3.1 Implementation of TANKS

TANKS was initially proposed in January 2017 by Byron Batteson [2] as part of his honors research project at the department of computing sciences at Nelson Mandela University. Three objectives were identified during the design phase of the project:

- apply sound pedagogical principles to the development of a coding game;
- be cost effective; and
- introduce introductory coding concepts mainly to primary school learners.

3.1.1 Pedagogical Principles

From the outset of the design process, it was decided that the features of the game as well as the targeted age groups needed to be founded upon sound principles. According to Piaget's theory of cognitive development [23–25] the *concrete operational stage* spanning the ages of seven to eleven years old is when children begin to perform mental operations such as problem-solving and arithmetic. The next stage of development is referred to as the *formal operational stage* around the ages of eleven or twelve years old when children can hypothesize different solutions to the same problem. Based on this, it was decided to set the target ages for the project at eight to twelve years of age.

A form of educational instruction called *scaffolding* is based on Vygotsky's theory and concept of the *zone of proximal development* [33]. This refers to the difference between a child's ability to learn when they receive guided supervision versus receiving no help at all. Scaffolding indicates that, although a child should receive supervision when learning, it should not be more than necessary. Scaffolding can be implemented by e.g., introducing levels of complexity to a game.

David, Triona and Williams [9] stated that a *hands-on* approach to learning could inform cognitive development via kinesthetic involvement, i.e. a transition from something concrete to something abstract. Rogers, Scaife, Gabrielli, Smith and Harris [29]

suggested that allowing children to use mixed realities (virtual or physical tools) within the context of play and learning allow for uncharacteristically extended interest and reflection. This extended interest would be key to an introduction to programming concepts. Making use of tangible tokens as part of TANKS is supported by these learning theories. Marshall [20] also stated that tangible programming may increase collaboration between children, with learning on a single desktop with a mouse and keyboard resulting in one or two children taking control of the application while others may only observe.

With these ideas in mind, it was decided that the games developed needed to include the features of problem solving (intrinsic to coding), scaffolding (levels of increasing complexity), physical interaction, and group work. Physical interaction could imply the use of robots, or the concept of tangible programming, where physical components are used to compile a set of instructions.

3.1.2 Evaluation of Extant Systems

Five different extant systems were investigated and evaluated based on the pedagogical principles identified above, their cost effectiveness, and how they relate to the introduction of coding concepts.

Logo and Lego MINDSTORMS. Seymour Papert [22] developed the Logo programming language for children in 1967, and in 1985 he began a collaboration with the Lego company to create Lego MINDSTORMS. Lego MINDSTORMS satisfies the objectives of applying pedagogical principles as well as introducing coding concepts, but is unfortunately not cost effective (see introduction of Sect. 3.1).

Algoblocs. Algoblocs is a programming instruction system [30] where large electrical components connect to each other, creating a set of logical steps that allow a computer program to operate. It has controls more closely related to typical programming languages and supports a collaborative learning experience. However, it is expensive due to its use of electronic components, and how it requires a computer to run its programs.

Tern. Tern was introduced in 2007 [16] as an inexpensive coding tool that uses wood blocks the learner connects to form a program. An image is captured with a webcam from either a computer or a laptop, and the program is interpreted and executed within a virtual environment. The primary disadvantage of Tern is that it is a computer-based solution requiring a webcam.

Scratch. Scratch is a visual programming tool that was released in 2007 [28]. Scratch uses visual coding blocks on a screen to create an animation, game, or simulation. It is a vastly more powerful tool than most tangible programming systems. However, it requires a computer and does not encourage group work.

Sheets. Sheets is a promising tangible programming system developed by Tada & Tanaka [31]. It utilizes printable paper-based command objects instead of the typical command blocks and is thus a more inexpensive successor to Tern. Sheets programs are also interpreted by a computer via a webcam, requiring the need for a computer.

After evaluation of these extant systems, it was decided that the TANKS game would contain the following features:

- Allow for the construction of code, including basic introductory coding concepts such as decision making, loops, and nested constructs;
- Make use of printable tokens to construct the code; and
- Apply image recognition in a mobile application that would internalise the tangible code that would then be executed on a standard smart phone.

Since smart phones are generally available to and owned by teachers and learners, it was concluded that this would be a cost-effective way to introduce basic coding concepts without the use of desktop or laptop computers. The printable tokens would support group work. To support problem solving and scaffold learning, the game would need to have levels of increasing complexity which would add an additional feature of gamification.

3.2 Game Rollout

By the end of the research project, the TANKS game was ready to be rolled out to schools to introduce learners to computing concepts. The game's usefulness and financial sustainability were critical for this to become a reality.

3.2.1 Usefulness

To determine the usefulness of the app, nine participants took part in a formal user evaluation [2]. The results showed that all participants successfully completed the sequence-oriented levels, but were not comfortable transitioning to the more advanced tokens. Many of the participants struggled with the interpretation of the "turn left" and "turn right" command tokens, as well as whether the command was relative to their perspective or the tank's perspective. Another common concern was the direction of the arrows on the "move forwards" and "move backwards" command tokens; these had to be adapted to become less ambiguous.

Some of the participants found capturing the image of the command token sequence to be unreliable, especially among low light, multiple lights, or surfaces giving off a glare. Overall, all participants found the game enjoyable and challenging.

As a result of this initial evaluation, some of the features of the application as well as the design of the physical tokens were improved. In other cases, it was noted that certain information would need to be explained prior to starting a TANKS workshop. Some of the most important aspects included creating a correct environment (e.g. background surface and lighting), taking proper smartphone photos (e.g. distance from the tokens, keeping the smartphone vertical and stable), as well as explaining the meaning of the basic commands and how they interacted with the tank in the game itself.

3.2.2 Financial Sustainability

It was decided at the outset that the business model would mainly identify corporate sponsors, whose funding would allow for the free distribution of the games to deserving

learners. This made it critical to get corporations on board to buy into the vision. In their relational framework of cause drivers for crowdfunding, Haasbroek and Ungerer [15] identify motivating factors such as making a difference, serving the greater good, and believing in the respective cause.

The first two years were used to reach as many learners as possible through direct workshops, while building up a storyline that would be attractive to media platforms. Having stories to tell and getting them told to the public became the main currency of the project during this stage. The stories told about learners enjoying the game in workshops, and the positive energy which was created regarding coding and solving problems. It was also important to show that learners from all kinds of communities were participating – especially those from disadvantaged townships and remote villages. Along with the excitement about programming came the additional educational aspects such as group work, problem solving, communication, strategizing, and dealing with mistakes.

Getting these stories told was critical for generating buy-in from corporate sponsors as well as the public. A wide range of media was utilized throughout the project. It was important to build up a positive relationship with journalists in the traditional media (print, radio, and television), and be in-tune about which stories would work for which media. Social media (Facebook and LinkedIn) became the most successful tools for giving the project momentum in the public space. LinkedIn was found to be the most effective channel for the more formal recruitment of sponsors. A valuable instrument became publishing own articles on LinkedIn, which provided more content than simple posts. While social media posts received more feedback than coverage in traditional media, traditional media is still viewed as more credible, which is why reposting traditional media coverage on social media was used as an additional strategy.

The best-accepted stories were those indicating the high quantities of learners that could be reached in schools without computers [35]; those about individuals that responded positively, and who achieved new dreams for their lives [13, 14]; as well as individuals and school teams that successfully participated in coding tournaments [19].

Although a positive public image is important for corporate sponsors, it was also important to address their more formal business expectations within this project. To do this, a partnership was formed with the Leva Foundation (a registered NPO) that could provide official Section 18A Tax Deduction Certificates for those contributing to these kinds of projects. Furthermore, documentation regarding the fact that mainly disadvantaged communities were reached was also important in terms of companies' BEE scorecards. Attempts were also made to tap into companies' skills and enterprise development CSI funding, albeit not with much success since TANKS programs to date have yet to be officially accredited.

3.2.3 Reflection

By the end of 2019, TANKS workshops had been presented to over 20,000 learners in all nine South African provinces. This was made possible through increasing buy-in from sponsors as well as the training of workshop facilitators in different regions. One of the highlights included a sponsored project in five provinces which culminated in a national event with 100 learners from 20 schools at the Sandton Convention Centre.

As part of Africa Code Week, over 70 facilitators were used in October 2019 to reach 11,000 learners in the Nelson Mandela Bay Municipality [35].

In addition to building relationships with sponsoring and media partners, the role of implementation partners became apparent. The Africa Code Week project was a partnership with a local company as well as the Nelson Mandela Bay Science & Technology Centre in Kariega [17]. Other implementation partners included the Govan Mbeki Mathematics Development Centre at the Nelson Mandela University, as well as Johannesburg libraries [27].

Although this rollout stage was successful in many aspects, as discussed in this section, challenges were also identified. Contrary to expectations, many learners, especially in rural settings, did not have phones that were modern enough play the TANKS game. Consequently, it was realized that many of the games distributed were not used at all. It thus became clear that corporate sponsorships were not well-utilized. A decision was therefore made to start focusing on schools, which resulted in the compilation of a set of tools in addition to the TANKS game.

3.3 Expanding the Project

The initial stage of interacting directly with learners was successful in giving momentum to the project, and simultaneously created challenges regarding its impact. The schools' focus necessitated the expansion of what we offered in the form of coding kits. Teachers for instance were interested in more than just a game and wanted something more formal that would effectively fit into or complement a curriculum. In addition to programming, a need was also identified to introduce computational thinking activities. And since coding is not yet officially part of their curricula, alternative ways of incorporating coding had to be explored.

3.3.1 Coding Kits

The first obvious need was for the compilation of a comprehensive TANKS coding kit that would empower teachers to introduce learners to coding. The main components of this kit are lesson plans, instructional videos on how to play the game, a solution set for the levels, and printed levels.

For the development of the set of seven lessons that could be used by teachers in class, the technology teacher Ms. Kelly Bush was approached to develop lessons that consecutively map onto every five levels in the game [3]. The following coding concepts are progressively covered here: sequential movement commands, "for" and "while" loops, optimizing code, and "if" statement decision making. Real life scenarios and physical activities are used in each lesson to introduce the different coding concepts to learners before they interact with the game.

The initial idea was that coding kits would be distributed to schools following a coding workshop attended by learners and teachers at the school. It was soon recognized that these workshops were not adequate to guide teachers through the more complex levels of the game, even though the concepts were also discussed in the lessons. In turn, eleven short instructional videos were compiled that not only introduced concepts to teachers and learners, but also assisted with issues such as troubleshooting when playing,

as well as how to organize a workshop. A further important feature of the videos was that they be small enough to send via WhatsApp to learners or parents requesting them.

It was noticed during the roll-out phase that, with some of the solutions that learners came up with, although they solved a problem, they were not examples of good coding. Consequently, a compendium of suggested solutions was provided for teachers to guide learners.

A coding kit typically allows for eight groups of five learners in a class to interact with the game. In a scenario where eight phones are not available, printouts of the 35 levels are provided to the groups. They build their solutions with the tokens, with the handful of "roaming" phones in the classroom used to evaluate the solutions and provide feedback. A further advantage of providing printed levels is that teachers can better check/count how many levels are completed per lesson.

3.3.2 Computational Thinking Activities

The term "computational thinking" was first applied by Seymour Papert, and later popularized by Jeanette Wing [22, 34]. Wing described it as the thought process involved in formulating problems and their solutions. It is generally accepted that any introductory coding module at the school level should include computational thinking exercises and activities.

There are numerous resources for computational thinking exercises [12, 18, 21] which are of great value to teachers. Some of them include Code.org, CS Unplugged, and the IITPSA's Computer Olympiad. As an additional tool for the coding kits, Keith Gibson from Collegiate Girls' High School in Gqeberha developed a compendium of 40 computational thinking activities [10, 11]. This compendium is a work in progress, and more activities will continuously be added. Kelly Bush also developed a compendium of similar activities for Foundation Phase learners [5, 6].

3.3.3 Alternative Ways of Incorporating Coding

The Department of Basic Education [10] currently envisions the implementation of coding and robotics in primary schools from grade R upwards. Draft curricula were published for this in March 2021, with specific schools identified as pilot schools. This will however only become official at the earliest in 2024. Although many teachers are optimistic about the idea of incorporating coding at school, coding activities often do not occur due to the reality of other formal subjects having higher educational priority. There are several alternative options being experimented to address the lack of time for this topic during normal school hours:

Incorporating Coding into Other Subjects. In support of previous observations by educators [8], many teachers have provided feedback regarding the relevance of coding activities regarding other curriculum content. For Foundation Phase teachers, because coding is related to spatial orientation, it becomes natural to incorporate these spatial activities into teaching on concepts such as left, right, below, on top, etc. Coding is often introduced as a list of instructions which can easily contribute to lessons related to verbal communication. Teachers of older learners report that they have used coding concepts as part of mathematics lessons with positive results.

Coding Clubs. Learners voluntarily participate in other sports or cultural activities at school. Coding clubs typically take place after school, during breaks, or on weekends [13, 14]. Encouraging feedback has been received by clubs, and described as learner-driven, with the learners watching the instructional videos while they progress through the coding levels. There have also been examples of coding clubs organized by NGOs that, although not directly linked to schools, have also been successful in reaching learners.

Coding Evangelists. Teachers are often overburdened and have neither the time nor the energy to teach extracurricular coding. In certain communities it has been observed that young people (not formal staff members) introduce learners to the TANKS coding project after receiving training in how to use the game. These individuals are either compensated by the school or are part of an industry- sponsored outreach. In addition to providing a service to the school's learners, this coding evangelist concept has great potential when it comes to employment opportunities for young people.

Coding Tournaments. Because the TANKS game was designed with levels of increasing complexity, it lends itself perfectly to competitive tournaments. This principle of gamification feeds into some learners' natural competitive nature, and results in them taking control of the learning process [3]. Various tournaments have taken place regionally and nationally with great success [28]. In preparing for these tournaments, participating teams put extensive effort into finding solutions to the challenges they will encounter in their competition.

Virtual Tournaments. Due to the sometimes difficult and expensive logistics of getting teams to a shared venue, the concept of virtual tournaments is being explored, most notably considering the increased travel obstacles during the COVID-19 pandemic. The BOATS coding app, developed by the same team that produced TANKS, was adapted for learners to interact on the app from anywhere, while scores are submitted to a central database. Several virtual tournaments have taken place, with over 1,000 learners from a few hundred schools participating [19]. A current work in progress is an honors research project involving a platform that will support virtual coding tournaments for the TANKS game [1].

3.3.4 Reflection

The above-mentioned approaches have all contributed to TANKS' impact, helping it evolve into a broader-based initiative. It has now taken on the characteristics of an unplugged coding movement throughout the country. This is an umbrella term referring to any methods used to introduce learners to coding without the use of computers. Since TANKS uses only mobile phones, we take the liberty of including it under this term. In addition to TANKS, various physical, pen-and-paper, and computational thinking activities are available to teachers. Despite the extensive efforts taken here including clear lesson plans, instructional videos, and launch workshops, there were still teachers who unfortunately did not have the confidence to start this kind of project at their respective schools. This made it clear that additional training was needed.

3.4 Teacher Training

The training of teachers was implemented in 2021 over three phases: introduction to unplugged coding, physical workshops throughout the country, followed by repeating the content of the physical workshops via four online sessions. For all training activities, the assumption was made that teachers had no prior coding-related training or experience. These training interventions formed part of an international collaboration between academics from South Africa, Namibia, Mozambique, and Germany under the broader topic of resilient communities. The collaboration was referred to as the YEEES project, which was sponsored by the German Academic Exchange Service (DAAD) and the German Federal Ministry of Education and Research [36].

3.4.1 Introduction to Unplugged Coding

Acknowledging that the concept of unplugged coding is unknown to educators in South Africa, a series of online workshops was presented by educators who are actively involved in using these techniques. The topics of the workshops included:

- Laying good foundations – coding concepts in the early years [4];
- Unplugged coding tools [18];
- The need for computational thinking skills in the teaching and learning of coding [21]; and
- Developing problem solving skills informally [11].

Over 500 educators registered for the workshops that were presented over four weeks.

3.4.2 Physical Workshops on Unplugged Coding Activities

As a direct result of the four online workshops, requests from educators were for physical workshops where they could be practically introduced to the various activities referred to in the initial workshops. Here, 12 workshops were presented across South Africa and Namibia, and attended by nearly 200 educators. The main topics that were covered within four hours were:

- What is coding – introduced by practical activities;
- The importance of computational thinking activities; and
- Using the TANKS coding app to introduce different coding concepts.

Of the 192 people that attended, 105 were teachers, 49 were NGO workers, 28 were staff of the Department of Basic Education, and 10 were academics from universities. The attendance of Basic Education staff was relevant considering how the draft curricula for Coding and Robotics for Primary Schools was published during this period.

From the 58 participants who responded to a survey, the following responses were given as reasons why coding was not offered in their schools: no computer, no trained teachers, not in the curriculum and not enough time in the day. All the responses were positive (100%) regarding the usefulness of the workshops. The comments included aspects such as: coding was demystified, realization that technology was not needed,

provided self-confidence to start with something, and realizing that their learners would enjoy the activities. These responses provide confirmation that the project was achieving the original objectives of the TANKS app: introduce computer programming in schools without computers, in a cost-effective way, and as a fun activity.

3.4.3 Online Workshops on Unplugged Coding Activities

Due to limited resources and time, the same content from the physical workshops was presented over four online workshops, with around 100 educators attending. All participants received the physical TANKS tokens in advance, along with electronic documents that were needed for some of the other activities. These online workshops expanded their potential reach, and opened doors for similar online workshops on various other platforms.

3.4.4 Reflection

While most participants had no prior coding experience, those attending the unplugged coding workshops found them useful. For the larger project, these workshops proved invaluable in motivating educators to get started with this kind of project by removing many of the perceived stumbling blocks. A challenge that remained for some of the delegates was the lack of time in their school day. Within the context of COVID-19 regulations, and some learners not being at school all week, this was unfortunately one of the realities we could not effectively address.

4 Guidelines

This section uses the experiences gained since 2017 (as discussed in Sect. 3) to present guidelines for introducing learners to computer programming in a developing country.

In the years since 2017, this TANKS unplugged coding project has evolved to entail more than a simple game. It continues to be supported by a successful financial model and is accompanied by effective training methods.

Rolled out in South Africa, this project is of significance for any developing country wanting to introduce its learners to coding. We provide guidelines regarding tools on how to use it and overcome any challenges that arise, alternative methods for its application, finances, and training.

4.1 Tools

Most developing countries face the reality that many of their schools lack computer laboratories, and simply lack the budgets to purchase expensive equipment such as robots. This means that the tools that are available cannot rely on access to computers or other equipment that is not readily available. It is assumed that schools and teachers do in fact have access to mobile phones. And learning activities can furthermore be supplemented by physical activities involving objects found in any classroom, as well as pen-and-paper activities.

In addition to being cost effective, tools need to apply pedagogical education principles such as problem solving, scaffolding, physical interaction, and group activities (see above). It should also be fun for the learners, taking advantage of gamification thinking [3].

Since teachers are used to formal curriculum-related activities, it is important to provide well-structured lesson plans involving the above-mentioned activities which interact with the tools provided. These lesson plans can be supplemented by short instructional videos related to their use.

Coding relies on the ability to solve problems, which means that computational thinking activities need to be provided to teachers, or they alternatively need to be directed to internet sources where these activities can be found.

4.2 Alternative Methods

Coding can often not be included in the official curricula of schools in developing countries, as a result creating the challenge of finding enough time to spend on coding during a normal school day. The first innovative way to address this is to find a relationship between coding and other curriculum content and incorporate coding into this instead. As an alternative, coding could be viewed in the same manner as sports and cultural activities, with the establishment e.g. of after-school coding clubs.

To relieve teachers from an already full schedule, coding evangelists (who are not part of staff) might represent an effective option and could rotate among schools. A further advantage of these evangelists would be the creation of employment for youth from local communities. A way to make coding activities exciting and encourage learners to participate would be to arrange coding tournaments for schools in the region [26]. Platforms allowing for virtual tournaments could extend their scope.

4.3 Training

Although the tools provided should always be as user-friendly and intuitive as possible, it needs to be acknowledged that most teachers are expected to introduce learners to concepts about which they have no prior knowledge. The concept of coding may be foreign to them, accompanied by a lack of technology to teach it in the first place, resulting in them having little confidence in even considering getting involved.

Training is required here, in either physical or online form. This training cannot assume any prior coding knowledge and must be interactive, with the teachers themselves having a positive, fun experience during the training. Presenters of the training must guard against making training too theoretical or making the teachers even more apprehensive by setting unreasonable goals.

4.4 Funding

It can generally be accepted that the schools most typically benefitting from the project will be those with limited financial resources. This means that the project's necessary tools and training must be made available to these schools at costs that are as low as

possible. On the other hand, for the project to be sustainable over a long period, it needs to have finances available to develop new resources, maintain current tools, and provide quality assistance that includes training and mentoring.

For this model to be effective, the drivers of the project need to initiate and maintain healthy partnerships with corporate sponsors and their corporate social investment budgets. Sponsors finding value in this support means that they will need/look for stories that prove they are making a difference. The effective use of media is essential in achieving this, whether this is traditional media or social media platforms. A third key partnership category includes implementation partners. These include schools and other entities that could make effective use of sponsorship. On a practical level, it is important for companies to obtain tax benefits from their efforts while improving their social impact scorecards. The project as a result should preferably be driven by a registered non-profit organisation.

5 Conclusions and Future Research

Many schools in developing countries have inadequate resources to address today's challenges. One of these includes preparing learners for the job market of Fourth Industrial Revolution. In South Africa, the government is driving the introduction of coding and robotics in all schools. The problem with this vision is that most schools do not have computer laboratories, and most teachers have no prior coding training. This reality in South Africa was the motivation for the development of unplugged coding concepts.

This paper reported on the TANKS unplugged coding project that was successfully rolled out in South Africa starting in 2017. From experiences gained during this rollout, generic guidelines covering the categories of tools used, finances, alternative methods, and training have been provided. Given the focus of introducing learners to coding, it is not planned to implement further coding concepts. Developing a "tournament app" which allows for the creation of new levels will be a great improvement. Further research could investigate the longer-term effects on educational outputs and teacher training identified over the course of time. The guidelines for the introduction of computer programming in schools in developing countries could contribute to helping achieve low-cost coding training within resource-limited contexts having limited available technology. Progress in this realm could make important contributions towards preparing youth for the challenges of the Fourth Industrial Revolution.

Future research will include the process of upscaling the project by reaching all the schools in South Africa. Furthermore, the steps needed to rollout the project in different countries across Africa need to be investigated through different case studies.

References

1. Adams, T.: An Investigation into the Facilitation of a Coding Tournament Across Different Venues, Using TANKS as a Case Study. Honours Treatise, Nelson Mandela University (2021)
2. Batteson, B.: Investigation and Development of an Inexpensive Educational Tool Suite for an Introduction to Programming. Honours Treatise, Nelson Mandela University (2017)

3. Buljan, M.: Gamification for Learning: Strategies and Examples (2021). www.elearningindustry.com. Accessed 4 Aug 2022
4. Bush, K.: TANKS Lesson Plans. Hudson Park Primary School, East London (2019)
5. Bush, K.: Laying good foundations – coding concepts in the early years. CODING UNPLUGGED Series, 3 February 2021 (2021a). www.yeees-project.org. Accessed 1 Mar 2021
6. Bush, K.: Computational Thinking Lesson Ideas. Hudson Park Primary School, East London (2021b)
7. BusinessTech: Here's how many South African schools don't have the internet or a computer lab – and what it will cost to fix the problem, 18 July 2018 (2018)
8. Catapult Learning: Why All Students Should Learn How to Code in Elementary School (2016). https://catapultlearning.com/2016/09/29/students-learn-code-elementary-school/. Accessed 3 Aug 2021
9. David, K., Triona, L.M., Williams, C.: Hands on what? The relative effectiveness of physical versus virtual materials in an engineering design project by middle school children. Cognition **43**(2), 1086–1109 (2006)
10. Department of Basic Education: Proposed Ammendment to the Curriculum and Assessment Policy Statement (CAPS) to make provision for Coding and Robotics Grades R to 9 (2021). https://www.education.gov.za. Accessed 3 Aug 2021
11. Gibson, K.: Developing Problem Solving Skills Informally. CODING UNPLUGGED Series, 24 February 2021 (2021a). www.yeees-project.org. Accessed 1 Mar 2021
12. Gibson, K.: Fun Computational Thinking. Collegiate Girls' High, Gqeberha (2021b)
13. Greyling, J.: Expanding Career Horizons in Rural Eastern Cape. Published on LinkedIn, December 2020 (2020a)
14. Greyling, J.: From TANKS in Grade 7 .. to a Computer Science PhD in the Future, March 2022 (2022)
15. Haasbroek, F., Ungerer, M.: Theory versus practise: assessing reward-based crowdfunding theory through a South African case study. South Afr. J. Bus. Manag. (2020)
16. Horn, M.: Tangible Programming in the Class with Tern (2007)
17. Kuli, S., Greyling, J., Stubbs, M.: TANKS and BOATS coding apps – indigenous tools bring coding and robotics to learners in disadvantaged school. In: SAASTEC Conferences, George, November 2019 (2019)
18. Makoena, S.: Unplugged coding tools. CODING UNPLUGGED Series, 10 February 2021 (2021). www.yeees-project.org. Accessed 1 Mar 2021
19. Malonde, Z.: Eastern Cape Pupils Shine in Coding Contest. The Herald, 29 July 2020 (2020)
20. Marshall, P.: Do tangible interfaces enhance learning? TEI **2007**, 163–170 (2007)
21. Oosthuizen, L.: The need for computational thinking skills in the teaching and learning of coding. CODING UNPLUGGED Series, 17 February 2021 (2021). www.yeees-project.org. Accessed 1 Mar 2021
22. Papert, S.: Mindstorms: children, computers and powerful ideas (1980). https://doi.org/10.1017/CBO9781107415324.004
23. Piaget, J.: The Origins of Intelligence in Children. New York: International Universties Press (1952a)
24. Piaget, J.: When thinking begins. In: The Origins of Intelligence in Children, pp. 25–36. International Universties Press, New York (1952b)
25. Piaget, J.: Intellectual evolution from adolescence to adulthood. Hum. Dev. **15**(1), 1–12 (1972)
26. Preller, C.: Coding ignites Madiba magic (2022). https://www.linkedin.com/pulse/coding-ignites-madiba-magic-jean-greyling. Accessed 4 Aug 2022
27. Sun, R.: City of Johannesburg celebrates coding by awarding skilled learners. Randburg Sun, 24 October 2019 (2019)

28. Resnick, M., et al.: Scratch: programming for all. Commun. ACM **52**(11), 60–67 (2009)
29. Rogers, Y., Scaife, M., Gabrielli, S., Smith, H., Harris, E.: A conceptual framework for mixed reality environments: designing novel learning activities for young children. Presence: Teleoper. Virt. Environ. **11**(6), 677–686 (2002)
30. Suzuki, H., Kato, H.: AlgoBlock: a tangible programming language, a tool for collaborative learning. In: Proceedings of 4th European Logo Conference, June 2016, pp. 297–303 (1993)
31. Tada, K., Tanaka, J.: Tangible programming environment using paper cards as command objects. Proc. Manuf. **3**(AHFE), 5482–5489 (2015). https://doi.org/10.1016/j.promfg.2015.07.693
32. The Conversation: Coding in South African schools: what needs to happen to make it work, 31 July 2019 (2019)
33. Vygotsky, L.S.: Play and its role in the mental development of the child. Sov. Psychol. **5**(3), 6–18 (1967)
34. Wing, J.: Computational thinking. Commun. ACM (CACM) **49**(3), 33–35 (2006). https://doi.org/10.1145/1118178.1118215
35. Willemse, N.: Coding without computers reaches thousands of learners. Mail and Guardian, 22 November 2019 (2019)
36. YEEES: Yields of Evocative Entrepreneurial Approaches on Environment and Society (2021). https://yeees-project.org/