Chapter 4
Engaging All Learners Through Quality Early Childhood Teacher Education

Lynn Ellen Cohen
Long Island University, USA

Angela Kalthoff
Capstone, USA

ABSTRACT

This chapter guides early childhood teacher educators, teachers, and policy makers to use appropriate technological tools in early childhood classrooms. It begins with learning theories related to computational thinking with a discussion of Seymour Papert and presents Marina Bers' framework. The chapter includes a professional development model for teachers and teacher educators. Technological apps and computational thinking are offered for use with face-to-face, hybrid, and remote instruction in early childhood settings. Many of the digital tools described in the chapter have numerous suggestions and resources available for teachers, teacher educators, and students to create a platform to begin using digital tools and evaluate how to select and use digital media. Changes can be brought about by reimagining classrooms and reshaping pedagogy with continued professional development.

INTRODUCTION

Technology plays a positive role in children's learning and development. In the 22nd century, early learners with advanced technological fluency may become the norm. The plethora of mobile apps, software, and stand-alone technologies intended for younger children tend to focus on teaching letters, numbers, and academic skills. Many educational agendas have seen attention directed to digital literacy and computational thinking in the last decade (Howland et al., 2018). Computational thinking involves "solving problems, designing systems, and understanding human behavior by employing analysis, abstraction, sequencing, negotiation, and consensus building technologies" (Chalmers, 2018, p. 93). While technol-

DOI: 10.4018/978-1-7998-6888-0.ch004

ogy uses are pervasive in ever changing digital contemporary society, young children's involvement in selected types of technologies requires intentional guidance to optimize cognitive, social, emotional, physical, and linguistic learning and development. Given the importance of academic content in early childhood programs and the importance of play for the young children's development, some digital tools can bridge academic content with playful and meaningful activities.

The National Association for the Education of Young Children (NAEYC) and the Fred Rogers Center for Early Learning and Children's Media (2012) delivered a joint position statement in 2012 outlining the ways digital learning is an inherent part of early childhood curriculum. The statement was intended to provide guidance to educators working in early childhood programs about the use of technology and interactive media. The statement discussed the delivery of technology tools in teacher education programs through professional development. Specifically, the NAEYC position statement states, " educators need the understanding, skills, and ability to use technology and interactive media to access information, communicate with other professionals, and participate in professional development to improve learning and prepare young children in a lifetime of technology use" (p.9).

More recently, Rideout (2017) shared that mobile media use and access is present in most households and with young children. Recently, Pila et al. (2019) released an updated report presenting trends on early childhood educators' access and use of technology since 2018. Pila et al.'s (2019) report not only examined the increase in mobile devices since the joint 2012 NAEYC/Fred Rogers position statement, they examined the extent to which educators and administrators felt they had enough technological support and professional development to implement technology in early childhood classrooms. Survey findings indicated since 2014, there has been little difference in the percent (49% v. 51%) of early childhood educators who report having professional development in technology. In addition, respondents felt a need to find and navigate digital media resources to make technology relevant and developmentally appropriate for young children (Pila et al., 2019).

As early childhood teacher educators, policymakers, and parents prepare for a future that is technologically progressing at a faster rate than ever before in a globalized world, the early childhood field needs to ensure that early childhood students can become effective problem solvers and critical thinkers. Our challenge in the field of early childhood is to equip children with the skills and aptitudes for the future and their everyday lives, without sacrificing the fundamental qualities of child development and children's play.

This chapter will guide early childhood teacher educators, teachers and policy makers to think about the use of appropriate technological tools in early childhood classroom settings. The chapter will begin with learning theories related to computational thinking with a discussion of Seymour Papert, known as the father of the computational thinking and present Bers' (2012) conceptual framework. Next, a professional development model for teachers and teacher educators is described. By bridging theory with practice, a description of several programs created for young children with the importance of understanding privacy policies of apps follows. Finally, the chapter concludes with implications for continued professional development and support to ensure technology is delivered to young children in intentional, playful ways.

BACKGROUND: THEORY OF CONSTRUCTIVISM AND CONSTRUCTIONISM

Two theories of learning and understanding that have influenced early childhood are *constructivism* and *constructionism.* They are sets of ideas to explain pedagogical approaches to teach students how to think computationally while programming (Ackerman, 2001; Bers, 2008; Papert, 1991; Papert, 1993; Piaget & Inhelder, 1969). They both involve the hands and the brain and "can enrich our understanding of how people learn and grow" (Ackermann, 2001, p. 1).

Jean Piaget named his theory *constructivism* because it referred to how learning occurs and how people construct knowledge because of constant interactions between the environment and the structures within the brains of the organism. "The constant balancing and re-balancing between mental structures of the learner is the result of active engagement with the environment and results in the construction of knowledge for that individual, who can act upon the environment in light of the new structures" (Waite-Stupiansky, 2017, p. 4). With a constructivist approach, learners synthesize their understanding and knowledge with real life experiences and reflect on them. Learners develop their own point of view and interpretation of the world. Constructivism is a theory of learning that builds upon a learner's prior experiences and knowledge (Piaget & Inhelder, 1969).

Seymour Papert worked with Piaget at the University of Geneva in the late 1950's and early 1960's. The collaboration led Papert's (1993) adaptation of constructivism to a theory he called constructionism. Papert's (1993) attention was the way children construct knowledge in the world with tangible hands-on objects as a support for constructions children create in their head. Constructionism is a theory of learning by doing, where the learner relies on tacit knowledge by programming on computers, tablets, program robots. It focuses on a belief that students learn best when working with other students, learning with other students and interacting with the real world. Constructionism brings both constructivism and programming language into the process of constructing knowledge and thinking computationally (Bers, 2018; Papert, 1991; Resnick, 2017). Papert (1991) did not define constructionism. He stated, "It would be particularly oxymoronic to convey the idea of constructionism through a definition since, after all, constructionism boils down to demanding that everything be understood by being constructed" (p.2). Constructionism empowers students in the use of programming language tools so they can design and create artifacts with programming tools.

In sum, both constructivism and constructionism have similarities and differences (Ackerman, 2001). The main similarity is that they both emphasize discovery methods of learning allowing learners to explore projects based upon their interests. Teachers facilitate and coach students while working on their projects to support, not dictate their ideas. The main difference between constructivism and constructionism is Piaget believed that learning is dependent on the development of mental functions, what happens in the mind as children create structures in the mind. Papert believed that learning depends on the development of physical objects with hands-on activities, epistemologically meaningful projects. For purposes of this chapter, the focus will be on Papert's constructionism as we describe appropriate technological tools in early childhood classroom settings.

The Positive Technological Development Framework

Papert (2000) claimed in his book, *Mindstorms: Children, Computers, and Powerful Ideas* was not a book about children and computers, it was a book about ideas. "My reference to *idea power* in *Mindstorms* was essentially positive: I wanted to show that some very powerful ideas could be brought into the lives of children through the mediation of digital technology" (Papert, 2000, p. 722). He sincerely wanted to give children a technologically rich environment to love ideas, be able to use ideas to solve problems, and make connections with other situations in the world for personal power. Papert's (2000) essay discussed that educators are faced with school curricula and did not have time for *big ideas.* Papert's constructionist approach had a vision of children becoming creators and producers of their own projects through programming.

Marina Umaschi Bers studied with Seymour Papert at the MIT Media Lab from the early 1990's until his injury in a traffic accident in Hanoi, Vietnam. Bers (2017) shared Papert's theory, philosophy, and pedagogy of power ideas with teachers and developed her own theoretical framework, Positive Technological Development (PTD). Marina's goal was to "translate Seymour's concept of powerful ideas to the world of early childhood education" (Bers, 2017, p. 14). The PTD framework challenges early childhood educators to think about students and how to help them become creators, innovators, problem solvers, critical thinkers, and collaborators (Bers, 2012). The PTD framework provides a model for designing and evaluating the use of new technologies. Previous early childhood studies including makerspace environments (Bers, Strawhacker, & Vizner, 2018; Strawhacker & Bers, 2018), KIBO robotics kit (Sullivan and Bers, 2018) have used the PDT framework.

Developmental science and developmental psychology informs PTD. Lerner's (2018) concepts and theories of human development focused on how to foster positive and pro-social development outcomes in children and young adults. Lerner et al. (2005) framed the developmental assets into six "C's" of Positive Youth Development (PYD): (1) caring, (2) connection, (3) contribution, (4) competence, (5) confidence, and (6) character.

Bers (2012) extended Lerner et al.'s (2005) research as a way to support positive behaviors through engagement with computers and innovative technologies. The PTD framework by Bers (2012) proposes six positive behaviors that include: (1) creation, (2) creativity, (3) communication, (4) collaboration, (5) community building, and (6) choices of conduct. It draws on Papert's (1980) big ideas related to constructivism and early research from the field of Positive Youth Development (PYD) (Lerner, 2018; Lerner et al., 2003). Below is the PTD framework with the six assets from PYD and six behaviors from PTD. The last column is related to classroom practices.

Six positive behaviors students' exhibit when engaging in developmentally appropriate digital explorations is evident in the PTD framework. Below are definitions of the six positive behaviors (Bers, 2012).

Content Creation. The opportunity to engage users in computer programming or computer applications that engage them in working with text, video, audio, graphics, and animations.

Creativity. The ability to transcend traditional ideas, rules, patterns, relationships, or interpretations and to create and imagine original new ideas, forms, and methods for using new technologies.

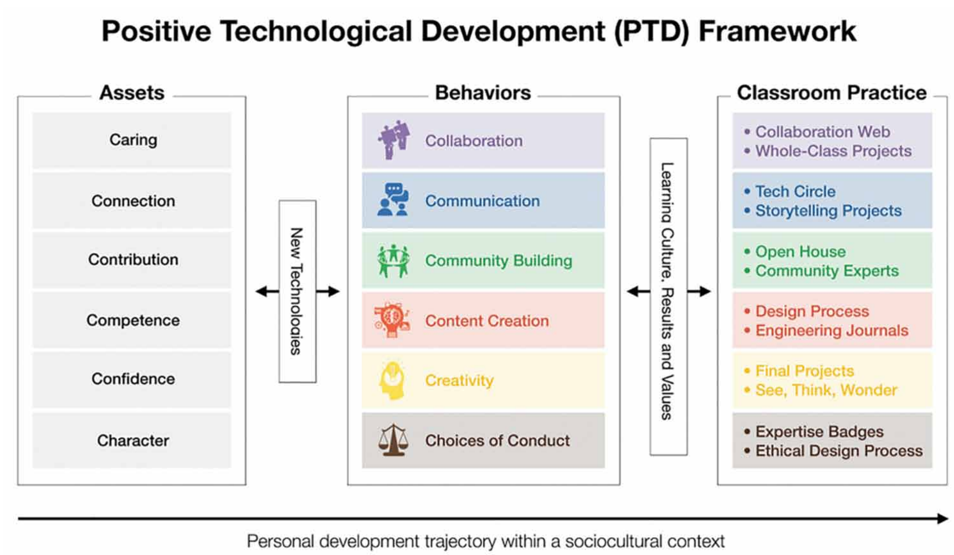
Choices of Conduct. The opportunity of making choices about our behaviors, explore "what if" situations, take action in the digital world, and experience the consequences.

Communication. The process of interchanging thoughts, opinions, or information by using technologies.

Collaboration. The opportunity to work with others and willingly cooperate toward a shared task. (Bers, 2012, p.11-12)

Bers (2018) developed a guiding framework that "integrates cognitive, personal, social, emotional, and moral development of early childhood. It is a curriculum of the powerful ideas of computer science that is developmentally appropriate." (p. 180). In the next section is a journey to learn and use powerful ideas with pre-service and in-service teachers.

Figure 1. Positive Technological Development © 2018, Marina Bers, used with permission



PROFESSIONAL DEVELOPMENT

Pila et al.'s (2019) report examined the extent to which educators and administrators felt they had enough technological support and professional development to implement technology in early childhood classrooms. The results indicated between 2014 and 2019, that only 2% (49% - 51%) of early childhood, educators reported having professional development in technology.

In 2018-2019, before COVID-19, social distancing and the use of masks, both authors participated in The Tufts Early Childhood Technology (ECT) certificate program https://go.tufts.edu/ECT at Tufts University. The first author was a participant and second author co-program manager. The first author was aware of Pila et al.'s (2019) research and wanted to learn current technologies to benefit future teacher candidates. The Tufts Early Childhood Technology (ECT) certificate program is a three-course blended program with completion in one-year. It is designed for teacher educators, in-service educators, museum directors, and librarians working in diverse settings. The program is housed within the Department of Child Study and Human Development at Tufts University.

During the 2018-2019 academic year, the first two courses were completed online, and the final course was completed in person at Tufts University. During the pandemic in 2020, the in person course was taught online in the same format of the first two courses. The ECT program consists of short lecture videos, course readings with written reflection, online discussions, and open-ended projects. The open-ended projects incorporated project-based learning. With the push for promoting technological fluency at the K-12 level over the past two decades both national and intentional (Code.org, 2021, https://www.code.org/) the ECT program incorporated projects that students were able to work through in their course work and implement in their educational settings.

The first course provided foundational knowledge related to educational technology. Participants made and designed projects for children using a variety of tools and specifically the KIBO robotics kit (Bers, 2018) and ScratchJr https://www.scratchjr.org) Each ECT student received a KIBO robotics kit at the beginning of the semester. KIBO Dances from Around the World Project (Sullivan, 2019) is a noteworthy robotics program designed by participants. Students built a program for their KIBO robot and incorporated programming concepts to make the KIBO robot perform a cultural dance to a song of their choice. Specific programming skills taught in class were incorporated into the KIBO projects.

ScratchJr (https://www.scratchjr.org) is a collaboration between *Developmental Technologies Research Group* (DevTech), led by Marina Bers, *The Lifelong Kindergarten Group* at MIT Media Lab, and led by Mitch Resnick and the *Playful Invention Company.* "It was inspired by Scratch, designed for children ages 8 and up by Mitch Resnick. Bers and Resnick specifically wanted an age-appropriate programming language for children ages 5-to-7-years old" (Bers, 2018, p.117). Using ScratchJr, the ECT participants practiced coding sequences and coding stories.

The second course provided foundational knowledge in educational spaces for young children, ECT participants learned to design and reflect upon their classroom spaces and change environments into makerspaces. ECT students were introduced to evidence-based, theoretical, and practical tools to examine classroom spaces and environments. ECT students explored a variety of technology and STEAM inspired environments, both in person and virtually, including (1) makerspaces, (2) museums, (3) libraries, and (4) classrooms. Participants chose a new space or research topic to redesign for a final project. A few project highlights included: (1) an early childhood preschool director designed a makerspace pamphlet for families to design a space in their home for their child(ren) and (2) a presentation proposal for a school board for funding for a classroom redesign.

The program culminated with a one-week in-person Early Childhood Technology Residency. The residency was a combination of lectures, workshops, and interacting with faculty from the DevTech Research Group. Participants were provided hands-on curricular experiences with children at the Eliot-Pearson Children's School under the supervision of program directors.

The Early Childhood Technology Certificate Program was an opportunity for the second author to facilitate learning for educators from around the world in an online learning format, collaborate with researchers in the DevTech Lab, and work with educators at the Eliot Pearson Children's School. The first author gained confidence and knowledge to teach early childhood preservice teachers computational programming and engineering in early childhood foundational courses. Scratch Jr. and KIBO robotics (Bers, 2018) were integrated into lectures for undergraduate and graduate pre-service teachers. Below is a reflection from an early childhood pre-service graduate student related to ScratchJr.

Before working on this project for class, I had never worked with or heard of ScratchJr.

Once I downloaded the app and began to play around with it and become familiar with it I really enjoyed it. I also imagined how much children would love to play and make up their own stories and pictures through Scratch Jr. It is a great way to allow children to be creative, express themselves through storytelling, coding and work with others in the medium of technology. Technology has become such an apparent and overbearing part of people's day-to-day lives. We use technology in the work place, for pleasure, in school, etc. Introducing an application like Scratch Jr to children is an amazing way to familiarize them with hands-on creative learning.

Here is a photo of a student programming the wooden KIBO blocks. She is scanning the blocks with the KIBO body to tell KIBO what to do. After scanning, the wooden blocks KIBO will move to her commands!

As shown through these two examples, professional development for early childhood teacher educators conveyed a powerful message about the kinds of technological learning behaviors to support in-service early childhood teacher candidates. The next two sections are related to information privacy and standards.

Information Privacy

When picking technological resources to use with students, it is important to think about information collected, as well as how companies use program information. Outside of specific guidelines set by educational institutions, there are policies and frameworks to help educators examine technology resources. The Children's Online Privacy and Protection Act (COPPA), is a law regarding how programs (www.ftc.gov) can collect data and personal information from children under the age of 13. The Family Educational Rights and Privacy Act (FERPA) is a Federal law that protects the privacy of student education records. FERPA protects personally identifiable information (name, family member's name, date of birth, place of birth, from students' education records from unauthorized disclosure. Common Sense Media is an organization for families and educators to find reviews, objective advice, and other helpful information. Common Sense Media created a resource focused on student privacy, https://privacy.commonsense.org/, and created a program that evaluates privacy policies to bring transparency to many commonly used apps well as websites. In addition to privacy, the U.S. Office of Educational Technology (https://tech.ed.gov/earlylearning/principles/) advices adults pay attention to apps that offer in-app purchases and advertising and avoid using apps with these features. Many free apps have the opportunity to purchase additional features or content with in-app purchases. This can be distracting to young children and take them outside of the app environment. Some apps that have advertisements also include the ability to track user behaviors, which could lead to privacy and security risks. Students' privacy in an online world is important to consider while planning digital lessons. In addition to protecting student privacy, another topic to think about is technology standards and the importance of using standards when teachers or teacher candidates write lessons. In the next section, *The International Society for Technology in Education* (ISTE) standards will follow.

Technology Standards

Several organizations have developed technology standards for the use of technology. When planning to use technology in classrooms it is important to have a clear understanding of the learning goals. The International Society for Technology in Education (ISTE), (https://www.iste.org/standards/for-students), is a nonprofit membership association for educators focused on educational technology and have published standards for students. The ISTE standards focus on the following seven standards to put the learner first and empower their voice: (1) Empowered Learner, (2) Digital Citizen, (3) Knowledge Constructor, (4) Innovative Designer, (5) Computational Thinker, (6) Creative Communicator, and (7) Global Collaborator. The ISTE standards for students provide a path for teachers to implement and align student technology use and student learning outcomes (Miller, et al., 2020).

Figure 2. Photo of Long Island University student coding with KIBO, used with permission

When planning for teaching technological tools it is important to think about standards and learning goals before the technology tool. In the next section, several technology tools and *ISTE Standards for Students* (Brooks-Young, 2016) are described followed by vignettes of classroom practices.

THEORY TO PRACTICE

Early childhood in-service, pre-service and teacher educators have a variety of new technologies to integrate into the early childhood curricula. With a learning first, technology second mindset, the focus is on plugged (meaning that students are engaging with activities on electronic devices) and unplugged (meaning that students are engaging with activities do not require an electronic device). Below is the rationale and evidence base for using digital tools. Practical ideas and examples are presented to prepare teacher educators to use digital tools. The focus of the activities encourage students to be active creators of learning. The vignettes were based on the co-author's experience teaching technology to young children. Pseudonyms were used for all child vignettes. Educators are encouraged to think about the goals and standards before picking the technological tools they use with early childhood students.

Technology Tools for Learning Digital Literacy

Using technological tools, students can create digital books, read eBooks, conduct research, share videos with friends, and draw and animate cartoons. Below are descriptions, websites, and related research studies of six technology tools used in a kindergarten classroom to support students' inquiry based learning. They include: (1) Book Creator, (2) Pebble Go, (3) Flipgrid, (4) Hero Elementary, (5) Seesaw, and (6) Toontastic.

Book Creator (https://bookcreator.com) is an app designed by Red Jumper Studio. It enables students to create e-books using photos, video, text and make voice recordings. Students can access files from saved data, the web, or instantly create a file within the app. McLaren's (2015) action research project in an integrated school in Australia used Book Creator as an interdisciplinary tool approach to curriculum. Book creator was useful for teaching reading groups. "Book Creator enabled me to create eBooks using each group's focused texts" (McLaren, 2015, p. 10).

A Capstone product, PebbleGo (https://www.pebblego.com/) is a digital tool for young learners to conduct research. It is curriculum-connected for students to choose from subject areas such as animals, health, science, social studies, and biographies to conduct research. Every article is audio recorded by professional voice-over artists, allowing students to have confidence in navigating the app to learn about topics of their interest.

According to the Flipgrid website, (https://info.flipgrid.com/) the mission is to engage and empower every voice in the classroom or community through recording and short videos. Kindergarten students (Johnson, et al., 2019) utilized Flipgrid to explain results of a project-based inquiry to parents and community members. Miller et al. (2020) suggested using Flipgrid to create research projects that require higher-level thinking. "Teachers assign topics and then students can respond to the topic, view others' videos, and reply to peers in another short video" (Miller, et al. 2020, p. 62). Early childhood educators can use Flipgrid as formative and summative assessment tools to assess learning with short recordings produced by students.

Hero Elementary is a series of activities, "including interactive games, educational apps, non-fiction e-books, hands-on activities, and a digital science notebook" (Lilin et al., 2019, p. 1). Hero Elementary is funded by the U.S. Department of Education, the Twin Cities Public Television (TPT) Ready to Learn project (2020). Hero Elementary (https://www.pbs.org/parents/shows/hero-elementary) was created for children in kindergarten to second grade in after school programs. The goal was to examine the impact of Hero Elementary playlists on student science learning. Lilin et al.'s (2019) study reported, "students enjoyed playing the digital games, which directly reinforced science concepts covered throughout the playlists" (Lilin et al., 2019, p. 22). Teachers reported students' science attitudes and knowledge of science concepts increased.

Seesaw (2019) (https://web.seesaw.me) has seen an increase in usage since COVID-19 because it supports meaningful learning anywhere through a learning loop. The learning loop brings together the teacher, student, family, and peers. Seesaw is a free application "which enables pupils to capture and present their work in a digital platform, eases teachers' job in monitoring pupils' progression and allows parents to leave constructive comments and feedback on pupils' work" (Rou & Yunus, 2020, p. 2392). Rou and Yunus (2020) conducted a study using Seesaw with early childhood English Language Learners (ELLs). After using Seesaw, ELL students' literacy skills increased as well as their interest in reading.

Google (2017) took over Toontastic 3D (2017) (https://toontastic.withgoogle.com) with Andy Russell as the product lead. Toontastic is a storytelling and animation tool that empowers students to create their own cartoons and share their stories with other children around the world. Students can draw, animate, narrate, and record cartoons. Roswell et al.'s study (2018) shared a study in which a graphic artist supported students to take on transmedia work and produce a graphic story with animation. Students worked in small groups using Chromebooks and used Toontastic to animate and share with classmates and adults.

Use of Tools and Standards in Practice - Ikan

"Oh, it's beautiful!" Ikran squealed as she reviewed images of an artic fox in a PebbleGo article. The class was researching animals and Ikran's teacher decided to implement an inquiry-based learning project to honor each students' interest in a study of animals. She guided students through each step in the inquiry based research process and facilitated their learning. Ikan used a variety of resources including eBooks, PebbleGo, and Hero Elementary. She digitally collected her work and shared her learning in a digital portfolio using Seesaw and Flipgrid. The digital portfolio allowed Ikran's teacher, classmates and parents to understand what Ikran was learning by capturing her work digitally.

Ikran learned about topics of interest to her through inquiry-based research, facilitated by her teacher. After digitally collecting research on animals through the resources in Hero Elementary and PebbleGo, Ikran could publish a book using Book Creator or create an animation with Toontastic.

Ikran's teacher guided the project based on several *ISTE Standards for Students* including (1) Empowered Learner, (2) Digital Citizen, and (3) Global Collaborator. Ikan was an *Empowered Learner* by choosing topics and tools to conduct her research. By sharing her work in a community space with classmates and parents, she was learning to be a *Digital Citizen*. Sullivan (2019) recommends allowing students to work on personally meaningful projects when learning and using new technology. Instead of Ikran's teacher having learners create a digital artifact about the same animal, she allowed choice and agency. Hattie (2009) states, "the computer is more effective when the student, not the teacher, is in control of the learning" (Hattie, 2009 p.225). By sharing her work on community platforms, Ikran is a *Global Collaborator*. She connected with children from a variety of backgrounds in her classroom, which broadened her learning and understanding.

Technology Tools for Learning to Program without a Screen

Students can learn about computational thinking through a variety of plugged and unplugged technology tools. Unplugged programming is defined as the teaching of programming concepts without a computer device (Farber et al., 2017), and is often implemented through bodily activity or using different objects and apparatus. Various examples of activities and related research studies identified include: (1) Bee-Bot, (2) KIBO, (3) Code.org, and (4) Robot Turtles.

Robots Without Screens

Several robotic devices have been created for students to learn how to code. Some devices require the use of an additional device with a screen, (e.g., tablet or computer) for students to program the robot, while others do not require a screen. The Bee-bot and KIBO robots are robots that do not require the use of a screen.

Terrapin (1979) developed Bee-Bot (https://www.terrapinlogo.com) a robot, designed to look like a bee. It uses directional keys allowing children to enter up to 40 commands sending Bee-Bot forward, back, left, right, clear, and go. When used in a classroom setting, the Bee-Bot is a tool encouraging collaboration and communication. Children are encouraged to work together to program their robot on a mat or create their own obstacles with classroom objects which they place around the room (Kazakoff & Bers, 2014). As children think about systematic directions to program into their robot, they are building their understanding of sequences, which is foundational for many other skills including literacy, problem solving and mathematical concepts (Patterson, 2016).

KIBO is a screen-free robot kit made for children ages 4- to 7-year-olds (KIBO, 2013) (https://www.kinderlabrobotics.com). To program KIBO, wooden blocks with barcodes are placed together to create a program with a *begin* block at the beginning and an *end* block at the end. The blocks are scanned with a red light reading each barcode to program the robot. KIBO allows young children to become engineers by constructing robots using motors, sensors, and craft materials (Sullivan & Bers, 2018). "KIBO has a natural aesthetic, making it equally appealing to children of any gender" (Sullivan, 2019 p. 106). It offers a playful and tangible way for children to learn computational thinking skills by building and programming a robot.

Unplugged Activities

Code.org (2021) (https://code.org/) is a nonprofit with a goal that every student in every school should have the opportunity to learn Computer Science (CS). Unplugged resources are available for educators starting in kindergarten. *Move It, Move It* is an unplugged lesson shared by Code.org in the Computer Science Fundamentals (CSF) curriculum. The goal of *Move It, Move It* is for students to move to specific directions created by their peers. Children create a simple algorithm (set of instructions) to move a classmate through a series of steps to reach a predetermined end-point using their body to give the directions. During this activity, children learn how to find and fix problems in their algorithms while learning how to be specific in their instructions.

Shapiro (2014) created Robot Turtles (http://www.robotturtles.com) a board game sold by *ThinkFun*. Students create a program with cards with arrows and actions to move a turtle from one place on a game board to a predetermined end place.

When first introduced to students, Shapiro (2014) recommends that the board is clear of any obstacles. As students' progress through the game adults can introduce obstacles and action cards to increase difficulty. "Young children develop science skills and learning by engaging in experiential learning" (Shapiro, 2017 p. 20). While playing this game, students work through computational thinking practices with a special emphasis called out, "debugging" which is when students find and fix errors. They are able to do this by using the special bug token. When they realize they have made an error in their code, they simply tap their token and say, "bug" and fix their mistake. The game encourages a playful approach to fixing mistakes.

Use of Tools and Standards in Practice - Lily

"North, East, East, North, stop!" said Lily as she directed her friend across a grid taped to the floor. The activity is followed with high fives, cheering and clapping. Lily is learning foundational computer science concepts through an unplugged activity (Move It, Move It) in her kindergarten class. Later, Lily played a board game with her friends and they program a turtle around a board to a jewel using arrow cards (Robot Turtles). Finally, Lily's teacher introduced the class to robots, which they are able to program through a grid on the floor (Bee-Bot or KIBO). Lily transferred her knowledge and experience of programming while using a robot.

Programming activities that encourage sequencing and problem solving without devices and use tangible objects are known as "unplugged." Students are engaged in activities like singing, dancing, and playing cards without a screen (Bers, 2019). Saxena et al. (2020) found when unplugged activities are first taught, preschoolers "have a better foundation for cultivating CT in plugged contexts" (p.57). Unplugged activities help develop CT skills and take away the need to learn the skills specific to the technological tool or software technology (Patterson, 2016).

Four-year-olds can successfully create programs for robots and screen-based applications while learning concepts of engineering design (Sullivan, 2019). While programming robots (KIBO, Bee-Bot), early childhood students are learning a variety of academic skills (e.g., numbers, size, shapes) while also learning to socialize and collaborate. They are developing fine motor skills and hand-eye coordination (Sullivan, 2019).

When introducing a new app, Patterson (2016) recommends students learn the pair program. Students work with partners either picked by the teacher or self-selected, so they can watch and then program together. "Pair programming, an agile software development technique in which two programmers work together at one computer. The *driver* writes the code while the *navigator* reviews and advises on code as it is typed. The two programmers switch roles frequently" (Krauss & Prottsman, 2016, p.12).

Lily's teacher guided the project based on several *ISTE Standards for Students* including: (1) Computational Thinker, (2) Creative Communicator, and (3) Innovative Designer. Lily's teacher provided Lily with the opportunity to be a *Computational Thinker* by using the power of the technology to create and test solutions. In addition, Lily learned to be a *Creative Communicator.* She learned to communicate and express herself through a variety of tools and formats. As an *Innovative Designer*, Lily took ownership over the programs she created for both her partner and the robots.

Technology Plugged Tools for Learning

A study conducted by Saxena et al. (2020) described how unplugged lessons were introduced first to provide students with the vocabulary and context, followed by the plugged activities that allowed students to draw on their prior experiences. Teachers reported that by first providing students with an unplugged lesson, students were able to "apply the CT skills to the plugged activity" (Saxena et al., 2020 p. 62). Plugged activities, activities using a device, allow students to solve puzzles and create programs by snapping blocks together. They learn about computer science fundamentals in developmentally appropriate apps by creating programs to move characters across the screen. The act of snapping blocks together in a specific sequence to create a program of their choice is also known as coding. Below are descriptions, websites, and related research studies of three technology tools to support students' foundational computer science concepts. They include: (1) Code.org, (2) Codespark, and (3) ScratchJr.

Code.org's Course A was created for young learners in order to learn about computer science fundamentals through a series of online puzzles. Kalelioğlu (2015) found the course materials in the Code.org K-8 *Intro to Computer Science* "suitable for kindergarten through to eighth grade" (Kalelioğlu, 2015, p.202). In Course A, students are faced with a challenge and they need to create a solution for their on-screen character to solve the puzzle. They use blocks with arrows to create a program as they code their way through challenges while learning about: (1) Digital Citizenship, (2) Sequencing, (3) Loops, and (4) Events. After the final puzzle is completed, students are presented a certification of completion with suggestions for what to do next. Students either move on to Course B or try an additional app like ScratchJr.

Hosford and Shochet (2014) designed and maintain Codespark (www.codespark.com), a program that offers a completely word free environment for early childhood students to solve puzzles and program stories and games. Children interact in self-directed play as they create stories and solve a series of puzzles with *The Foos*. *The Foos* are a set of brightly colored cartoon characters with unique characteristics. Children program *The Foos* as they advance through a series of locations in *Fooville*. While programming the *Foos*, children are engaged in computer science foundational skills as well as developing literacy skills (Patterson, 2016).

The ScratchJr (2017) website (https://www.scratchjr.org) has a variety of teaching activities (drive across the city, run a race, play a game) that teachers can access with simple step-by-step instructions to guide children in selecting a background, selecting characters, and making programs to practice coding sequences and coding stories. Researchers have used ScratchJr in early childhood programs with preservice kindergarten teachers (Kalogiannakis & Papadakis, 2017; kindergarten, first and second grade students (Portelance et al., 2015); and third grade students (Chou, 2019). ScratchJr is an appropriate educational environment for pre-service early childhood teachers to learn programming basics.

Use of Tools and Standards in Practice - Vladimir

"Yes! I did it!" Vladimir is delighted as his character moves across the screen in a series of actions that he programmed. He used one of the block-based programming apps (Code.org, (code.org/), CodeSpark (https://www.codespark.com) to support computational thinking and build literacy skills through sequencing. Patterson (2016) categorized these apps in two ways: (1) *Leveled Apps* and (2) *Open Studio Apps. Leveled Apps* provide an environment for students to navigate characters through challenges using arrows and other visual commands in the form of blocks snapped together. *Open Studio Apps* provide an environment where students use the blocks to create animations or games. Students are using problem solving skills without solving puzzles or following specific rules.

Research (Bers, 2018; Lee et al., 2013; Portelance & Bers, 2015) has found peer collaborations and communication skills increase through coding activities. Lee et al. (2013) examined social interactions with kindergarten students using a structured versus unstructured robotics curriculum and found the less structured learn-by-doing approach was useful for teachers when integrating technology. Video-based interviews with pairs of second grade students provided an opportunity to share knowledge of ScratchJr programming projects (Portelance & Bers, 2015).

Vladimir's teacher guided the project based on several *ISTE Standards for Students* including: (1) Computational Thinker, (2) Knowledge Constructor, and (3) Creative Communicator. Vladimir's teacher provided him the opportunity to be a *Computational Thinker* by allowing him to develop and code programs. Vladimer completed Code.org's Course A solving puzzles, received a certificate, and then moved on to program a soccer game using Scratch Jr. When his attempts at programming were not correct, he continued to persevere to find solutions. Vladimir started with a goal and broke it down into smaller sequential sectors. Vladimir programmed his soccer game for several days. As a *Knowledge Constructor,* Vladimir explored his knowledge of soccer to program a game of soccer using ScratchJr. Vladimer's teacher assessed his social interactions by observing and recording his interactions with kindergarten peers while using the tools. As a *Creative Communicator*, Vladimer used expression and complete sentences to share his soccer program with other children in the class.

CONCLUSION

This chapter highlighted the benefits of using appropriate digital media to support the skills students need to function and flourish in a technological age. A theoretical framework designed by Bers (2013) was provided for early childhood researchers. Tufts University's ECT professional development certificate program models opportunities for teacher educators. Last, an extensive discussion of a variety of digital media and implementation in a kindergarten class were offered.

CHALLENGES AND RECOMMENDATIONS

Apart from garnering technological tools, processes, and ISTE standards associated with technology in early childhood, there are certain challenges and possibilities with implementation of a quality technological classroom. First, it is important to think about the criteria for choosing a tool and the learning outcomes for students. "The Positive Technological Development Framework attempts to provide a model for how development can be supported by the use of technologies" (Bers, 2012, p. 9). PTD builds on the competence and confidence in the use of computers. Educators should also consider the creative and communication advantages students acquire when using high quality interactive media. Bers' (2012) PTD is aligned with the *Framework for 21st Century Skills* (Partnership for 21st Century Skills, 2007). To obtain technological fluency, teachers need to help students become creators rather than consumers of technology. Thoughtful considerations of the student, curriculum goals, and standards are a starting point when using technology.

In 2020, a challenge of the COVID-19 virus arose for families, teachers, and schools. Educational institutions needed to figure out how to deliver education for the students they served. Some institutions were able to train teachers and launch remote learning. Services were offered to parents/caregivers as they scaffolded learning in their homes using tablets, laptops, PC's and smartphones (Dias, et al. 2020). Additionally, teacher education programs needed to consider new ways and skills to deliver knowledge for preservice early childhood teachers in order to deliver remote learning.

The digital tools offered in this chapter can be used for remote learning and offered a partial solution to supporting educators during the COVID-19 pandemic. Digital literacy and plugged learning tools are apps available to download on tablets, computers, and smartphones. KIBO, an unplugged tool is available for families to borrow for three months (https://kinderlabrobotics.com/sign-up-for-the-kibo-home-robotics-course). With any digital app, parental/caregiver guidance is necessary. Common Sense Media https://www.commonsensemedia.org is an excellent resource for teachers and teacher educators to share with parents/caregivers. The pandemic has demonstrated the importance of building knowledge, skills and abilities with technology. As we move forward into post-COVID-19 we need to focus on at-home technologies complementary to the school's curricula and developmentally appropriate practice for young learners.

Pila's et al.'s (2019) updated findings of NAEYC's 2012 position statement reported that teachers continue to feel they do not have enough technological support and professional development to implement technology and remote learning to early childhood students. Early childhood teachers often use technology in their teaching, but usually as tools to show YouTube videos or interactive white board technology. There appears to be a gap between courses they took at the university level and the level of Information Communications Technology (ICT) they are expected to use in their practice (Kalogiannakis, 2010). Teachers now need well-designed college and in-service training courses.

Langub and Lokey-Vega (2017) offered an interesting framework to improve digital literacy in a graduate early childhood education course using a design case. The redesign supported personal and professional development goals because course participants created a workable technology plan. Environments were created to "foster digital literacy in creative and collaborative ways that supported communication and critical thinking" (Langub & Lokey-Vega, 2017, p. 330). These are the same goals of the PTD Framework, *Framework for 21st Century Skills* (Partnership for 21st Century Skills, 2007) and *ISTE Standards for Students* (2017).

The Tufts University Early Childhood Technology (ECT) certificate program provided the authors opportunities to train teachers in the use of digital tools. Graduate students used technology tools in course work. As described in this chapter, newer digital media for using book making and coding tools are important for fostering technological competence in our young learners. Highlighted were vignettes of kindergarten students using technology tools (i.e. ScratchJr and Code.org) for use in early childhood programs. Many digital tools described in the chapter have numerous suggestions and resources available for teachers, teacher educators, and students to create a platform to begin using digital tools in early childhood programs. We invite you to begin educating yourself about appropriate resources and use the digital tools and the rationale for using the tools as you work to integrate new literacies into teacher education.

REFERENCES

Ackermann, E. (2001). Piaget's constructivism, Papert's constructionism: What's the difference? *Future of Learning Group Publication*, *5*(3), 438.

Bers, M. U. (2008). *Blocks to robots: Learning with technology in the early childhood classroom*. Routledge.

Bers, M. U. (2012). Designing digital experiences for positive youth development: From playpen to playground. Oxford. doi:10.1093/acprof:oso/9780199757022.001.0001

Bers, M. U. (2017). The Seymour test. Powerful ideas in early childhood. *International Journal of Child-Computer Interaction*, *14*, 10–14. doi:10.1016/j.ijcci.2017.06.004

Bers, M. U. (2018). *Coding as playground: Programming and computational thinking in the early childhood classroom*. Routledge.

Bers, M. U., Strawhacker, A., & Vizner, M. (2018). The design of early childhood makerspaces to support positive technological development: Two case studies. *Library Hi Tech*, *36*(1), 75–96. Advance online publication. doi:10.1108/LHT-06-2017-0112

Book Creator. (2011). *Remote learning with book creator*. https://bookcreator.com

Brooks-Young, S. (2016). *ISTE Standards for students. A practical guide for learning with technology*. D.C. International Society for Technology in Education., https://www.iste.org/standards/for-students

Chalmers, C. (2017). Robotics and computational thinking in primary school. *International Journal of Child-Computer Interaction*, *17*, 93–100. doi:10.1016/j.ijcci.2018.06.005

Children's Online Privacy and Protection Act. (1998). https://www.ftc.gov

Chou, P. N. (2020). Using ScratchJr. to foster young children's computational thinking competence: A case study in a third-grade computer class. *Journal of Educational Computing Research*, *58*(3), 570–595. doi:10.1177/0735633119872908

Code.org. (2021). *Learn computer science. Change the world*. https://www.code.org/

Common Sense Media. (2003). *Privacy program*. https://privacy.commonsense.org/

Dias, M. J. A., Almodóvar, M., Atiles, J. T., Vargas, A. C., & Zúñiga-Leon, M. (2020). Rising to the challenge: Innovative early childhood teachers adapt to the COVID-19 era. *Childhood Education Innovations*, *96*(6), 38–42. doi:10.1080/00094056.2020.1846385

Faber, H. H., Wierdsma, M. D. M., Doornbos, R. P., van derVen, J. S., & de Vette, K. (2017). Teaching computational thinking to primary school students via unplugged programming lessons. *Journal of European Teacher Education Network*, *12*, 13–24.

Flipgrid, Inc. (2014). *Empower every voice*. https://www.info.flipgrid.com/

Google Inc. (2017). *Toontastic* [Mobile application software]. https://toontastic.withgoogle.com

Hattie, J. (2009). *Visible learning. A synthesis of over 800 meta-analyses relating to achievement*. Routledge.

Hosford, G., & Shochet, J. (2014). *Codespark*. https://codespark.com

Howland, K., Good, J., Robertson, J., & Manches, A. (2018). Editorial. Special issue on Computational thinking and coding in childhood. *International Journal of Child-Computer Interaction*. Advance online publication. doi:10.1016/j.ijcci.2018.11.001

International Society for Technology in Education. (2017). ISTE standards for students. A practical guide for learning with technology. Washington, DC: Author.

Johnson, L., McHugh, S., Eagle, J. L., & Spires, H. A. (2019). Project-based inquiry (PBI) global in kindergarten classroom: Inquiring about the world. *Early Childhood Education Journal*, *47*(5), 607–613. doi:10.100710643-019-00946-4

Kalelioğlu, F. (2015). A new way of teaching computer-programming skills to K-12 students: Code.org. *Computers in Human Behavior*, *52*, 200–210. doi:10.1016/j.chb.2015.05.047

Kalogiannakis, M. (2010). Training with ICT for ICT from the trainer's perspective. A local ICT teacher training experience. *Education and Information Technologies*, *15*(1), 3–17. doi:10.100710639-008-9079-3

Kalogiannakis, M., & Papadakis, S. (2017). *Pre-service kindergarten teachers' acceptance of ScratchJr. as a tool for learning and teaching computational thinking and science education* (Paper Presentation). European Science Education Research Association, Bologna, Italy.

Kazakoff, E. R., & Bers, M. U. (2014). Put your robot in, put your robot out: Sequencing through programming robots in early childhood. *Journal of Educational Computing Research*, *50*(4), 553–573. doi:10.2190/EC.50.4.f

KIBO. (2013). Available online https://www.kinderlabrobotics.com

Krauss, J., & Prottsman, K. (2017). *Computational Thinking and Coding for Every Student: The Teacher's Getting-Started Guide*. Sage, Corwin Press.

Langub, L. W., & Lokey-Vega, A. (2017). Rethinking instructional technology to improve pedagogy for digital literacy: A design case in a graduate early childhood education course. *TechTrends*, *61*(4), 322–330. doi:10.100711528-017-0185-1

Lee, K. T. H., Sullivan, A., & Bers, M. U. (2013). Collaboration by design: Using robotics to foster social interaction in kindergarten. *Computers in Schools*, *30*(3), 271–281. doi:10.1080/07380569.2013.805676

Lerner, R., Lerner, J., Almerigi, J., Theokas, C., Phelps, E., & Gestsdottir, S. (2005). Positive youth development, participation in community youth development programs, and community youth programs, and community contributions of fifth-grade adolescents: Findings from the first wave of the 4-H study of positive youth development. *The Journal of Early Adolescence*, *25*(1), 17–71. doi:10.1177/0272431604272461

Lerner, R. M. (2018). *Concepts and theories of human development* (4th ed.). Lawrence Erlbaum Associates. doi:10.4324/9780203581629

Lerner, R. M., Dowling, E. M., & Anderson, P. M. (2003). Positive youth development: Thriving as a basis of personhood and civil society. *Applied Developmental Science*, *7*(3), 172–180. doi:10.1207/S1532480XADS0703_8

Linlin, L., Mahrer, B., Weiser, G., Chow, E., Orenstein, A., & Atienza, S. (2019). *Hero Elementary playlist pilot study in after-school programs*. WestEd.

McLaren, D. (2015). IPads in the bush. *Practically Primary*, *20*(1), 10–12.

Miller, S. C., McIntyre, C. J., & Lindt, S. F. (2020). Engaging technology in elementary school: Flipgrid's potential. *Childhood Education*, *96*(3), 62–69. doi:10.1080/00094056.2020.1766677

National Association for the Education of Young Children (NAEYC) & Fred Rogers Center for Early Learning and Children's Media at Saint Vincent College. (2012). *Technology and interactive media as tools in early childhood programs serving children from birth through age 8*. Washington, DC: NAEYC; Latrobe, PA: Fred Rogers Center.

Papert, S. (1991). Situating constructivism. In I. Hartel & S. Papert (Eds.), *Constructivism* (pp. 1–12). Ablex Publishing Corporation.

Papert, S. (1993). *Mindstorms: Children, computers, and powerful ideas* (2nd ed.). Basic Books.

Papert, S. (2000). What's the big idea? Toward a pedagogy of idea power. *IBM Systems Journal, 39*(3-4), 720-729.

Partnership for 21st Century Skills. A Network of Battelleforkids. (2019). https://battelleforkids.org/networks/p21

Patterson, S. (2016). *Programming in the primary grades: Beyond the hour of code*. Rowman & Littlefield.

PebbleGo. (2009). *Where students go to find answers*. https://www.pebblego.com/

Piaget, J., & Inhelder, B. (1969). *The psychology of the child*. Basic Books.

Pila, S., Blackwell, C. K., Lauricella, A. R., & Wartella, E. (2019). *Technology in the Lives of Educators and Early Childhood Programs: 2018 Survey*. Center on Media and Human Development, Northwestern University.

Portelance, D. J., & Bers, M. U. (2015). *Code and tell: Assessing young children's learning of computational thinking using peer video interviews with ScratchJr*. doi:10.1145/2771839.2771894

Portelance, D. J., Strawhacker, A. L., & Bers, M. U. (2015). Constructing ScratchJr programming language in the early childhood classroom. *International Journal of Technology and Design Education*, *26*(4), 489–504. doi:10.100710798-015-9325-0

Resnick, M. (2017). *Lifelong kindergarten. Cultivating creativity through projects, passion, peers, and play*. MIT Press. doi:10.7551/mitpress/11017.001.0001

Rideout, V. J. (2017). *The Common Sense census: Media use by kids age zero to eight*. Common Sense Media.

Roswell, J., Lemieux, A., Swartz, L., Turcotte, M., & Burkitt, J. (2018). The stuff that heroes are made of: Elastic, sticky, messy literacies in children's transmedial cultures. *Language Arts*, *96*(1), 7–20.

Rou, L. Y., & Yunus, M. M. (2020). The use of Seesaw in increasing pupils' reading interest. *Universal Journal of Educational Research*, *8*(6), 2391–2396. doi:10.13189/ujer.2020.080623

Saxena, A., Lo, C. K., Hew, K. F., & Wong, G. K. W. (2020). Designing unplugged and plugged activities to cultivate computational thinking: An exploratory study in early childhood education. *The Asia-Pacific Education Researcher*, *29*(1), 55–66. doi:10.100740299-019-00478-w

ScratchJr. (2017). *Learn Page*. Retrieved from https://www.scratchjr.org/

Seesaw. (2019). Available at https://web.seesaw.me

Shapiro, D. (2017). Robot Turtles: The game for little programmers. *Science and Children*, *54*(6), 20.

Strawhacker, A., & Bers, M. U. (2018). Promoting positive technological development in a Kindergarten makerspace: A qualitative case study. *European Journal of STEM Education*, *3*(3), 1–21. doi:10.20897/ejsteme/3869

Sullivan, A. (2019). *Breaking the stem stereotype: Reaching girls in early childhood*. Rowman & Littlefield.

Sullivan, A., & Bers, M. U. (2018). Dancing robots: Integrating art, music, and robotics in Singapore's early childhood centers. *International Journal of Technology and Design Education*, *28*(2), 325–346. doi:10.100710798-017-9397-0

Terrapin. (1979). *Beebot*. Available online from https://www.terrapinlogo.com

Toontastic. (2017). Available online from https://toontastic.withgoogle.com

Twin Cities Public Television (TPT). (2020). *Hero Elementary*. https://www.pbs.org/parents/shows/hero-elementary

Waite-Stupiansky, S. (2017). Jean Piaget's Constructivist Theory of Learning. In L. Cohen, S.W. Stupiansky (Eds). Theories of Early Childhood Education: Developmental, Behaviorist, and Critical. Routledge. doi:10.4324/9781315641560-1

KEY TERMS AND DEFINITIONS

Computational Thinking: Computational thinking involves solving problems, designing systems, and understanding human behavior by employing analysis, abstraction, sequencing, negotiation, and consensus building technologies.

Constructionism: Constructionism is a theory of learning by doing, where the learner relies on tacit knowledge.

Constructivism: Constructivism refers to how learning occurs and how people construct knowledge because of constant interactions between the environment and the structures within the brains of the organism.

Digital Literacy: An individual's ability to find, evaluate, and compose clear information through writing and other media on various digital platforms.

Early Childhood Technology (ECT) Graduate Certificate Program: A graduate program housed within the Department of Child Study and Human Development at Tufts University that was designed for educators working in diverse settings.

Pair Programming: Pair programming, an agile software development technique in which two programmers work together at one computer. The driver writes the code while the navigator reviews and advises on code as it is typed. The two programmers frequently switch roles.

Positive Technological Development Framework (PTD): A framework developed by Dr. Marina Umaschi Bers in which there is an emphasis to encourage students to become creators, innovators, problem solvers, critical thinkers, and collaborators.

Unplugged: Unplugged programming is defined as the teaching of programming concepts without a computer device.