Starting from Scratch: Using Scratch as a Montessori Material to Develop Digital Literacy

by

Simon Tsun-Hong Kwok

B.Ed., Simon Fraser University, 2008
B.Sci., Simon Fraser University, 2007

A Project Submitted in Partial Fulfillment of the Requirements for the Degree of

MASTER OF EDUCATION

in

The Department of Curriculum and Pedagogy
(Media & Technology Studies)
(Digital Learning & Curriculum Cohort)

The Faculty of Graduate & Postdoctoral Studies

THE UNIVERSITY OF BRITISH COLUMBIA
(Vancouver)

June 2017

Abstract

This qualitative study looked at how Scratch, a programming language created by the Massachusetts Institute of Technology, would help students develop digital literacy. In addition, since Montessori programs operating in public schools will also be adopting this new curriculum, this study suggested that Scratch be considered a learning material that could become part of the Montessori environment. The study took place over twelve months, with data collected from two participating middle school classes (Grades 6-8, mixed gender, with a variety of digital skills). One class composed of 18 participants in a Montessori program, while the other consisted of 26 participants in a neighbourhood program. Participants explored coding with Scratch, first going through self-guided tutorials then proceeding to design and create a program (e.g., game, animation, or app) of their choice. Data were gathered throughout the process, which included in-class observations and design notes compiled by participants using Microsoft OneNote. At the end of the project, participants provided their experience with using Scratch through informal small group conversations as well as surveys. The findings of this study were supported by a comprehensive literature review of key concepts such as digital literacy and the Montessori philosophy, combined with teacher observations and student accounts of their experience with using Scratch for learning. The data collected showed that Scratch not only enabled students to develop digital literacy based on ISTE's achievement indicators (2016), but also met the general criteria outlined by Prozesky and Cifuentes (2014) for a technology to be integrated into the Montessori environment. Although Scratch is the primary subject of analysis, future studies could employ a similar review method to determine whether other coding applications could be considered learning materials for Montessori students to develop digital literacy.

Keywords: Scratch, computer programming, Montessori material, digital literacy

Table of Contents

List of Tables

List of Figures

Acknowledgements

I would like to offer my gratitude to my professors at UBC who taught me over the past two years, especially Dr. Stephen Petrina, Dr. Paula MacDowell, and Dr. Rachel Ralph for the continuing support and inspiration on my graduate project as well as to become a better educator. They have challenged me in every aspect over the course of the master's program, and although there were speed bumps and obstacles, they provided guidance patiently by answering many questions that were thrown their way.

I would also like to thank my critical friends in the DLC3 cohort who had provided me with numerous thought-provoking ideas during our class discussions that helped move my research forward. I especially appreciate Belinda Scott and Cherie Nagra with whom I spent hours and hours working with on various assignments and presentations. They have given me invaluable feedback and suggestions every step of the way in my inquiry and have made this two-year journey a fun and memorable one.

For my parents, Anthony and Susanna, for the support they have given me over the past two years both emotionally and financially. Thank you for putting up with my erratic work schedule and keeping me well-fed physically while I fed myself intellectually.

Introduction

The 1990s saw a boom in the development of digital technology and entertainment, which just happened to coincide with the period in which I grew up. One of the biggest technology developments at the time was the production of video game systems, and just like any other boys who grew up playing video games, my career goal at that time was to become a successful video game tester. I kept thinking to myself, *“Who wouldn’t want to make a living out of playing video games all day, every day?”* I eventually grew out of this childish and naïve goal but continued to work towards my aspirations by taking ICT courses in middle school as well as computer programming and drafting and design classes in high school. By that time, I came up with a more realistic goal of enrolling into a post-secondary Computer Science program with the hopes to work for IT companies since I was always curious about digital technology and enjoyed learning how to create and develop something that could be used by other people, whether it be games or applications. At that time, the thought of being a teacher never entered my mind, but after developing an interest in working with children and youth in my university days, I decided to forgo my original plan of working in the IT industry and applied into the Professional Development Program to become a teacher.

Even after switching to a different career path, I remained passionate about integrating digital technology into my practice as an educator. This passion grew even more when I became a Montessori teacher, as I could not help but notice how little digital technology is used in Montessori classrooms to develop 21st century learners at the middle school level. Although I have explored Scratch — a multimedia programming language developed at the MIT Media Lab — earlier during my teacher training, it was more to establish a connection to the constructivist theory of learning. After being reintroduced to the program at a seminar, I became very curious about how learning to code with Scratch could help middle school students develop digital literacy, one of the key competencies of 21st century learning. Moreover, I wondered if Scratch could be considered a digital learning material that is part of the Montessori prepared environment.

Project Description

> *"If you learn to code, you can code to learn…When you learn to code, it opens for you to learn many other things." –Mitch Resnick (2012)*

These words resonated with me so profoundly during Mitch Resnick's talk at the TEDxBeaconStreet event in November 2012. People may interpret this quote as nothing more than an advertisement to promote computer programming; to me, programming is only part of a bigger picture called computational thinking. More intriguingly, computational thinking is not only about thinking about computers. It could be applied to a multitude of disciplines and its associated skills are highly transferable (Krauss & Prottsman, 2017; Wing, 2006). This is what Resnick is alluding to when he spoke about coding, and is considered one of ISTE's (2016) seven achievement indicators of digital literacy.

We live in the world that is populated by young people who are born, learn, and develop while immersed in the digital world. Just as one the primary goals of traditional education is to allow students to become literate, it is equally important for them to become *digitally* literate. With the push for 21st century learning, the introduction of the new ADST curriculum in British Columbia, and initiatives to encourage students to bring their own electronic devices into the classroom, I felt that I would do my students a disservice if I did not teach them digital literacy skills.

While multiple foci of the Montessori method such as self-direction (P. P. Lillard, 1996, 2011), life-long learning (American Montessori Society, 2015), peace and global citizenship (Duckworth, 2006) align with the 21st century competencies suggested by C21 Canada (2012), the incorporation of digital technology use in the Montessori classroom has been very limited. Various studies in recent years began to look at using technology in a Montessori classroom, but they were mainly focused on being used for word processing (Dunn, 2000), organization and research (Hubbell, 2003), and to assist students with learning difficulties (Boyd, 2008). Therefore, I planned to explore new ways in which digital technology could be used in the Montessori classroom to develop digital literacy — a 21st century competency that Montessori education has yet to address.

Since Scratch uses a constructivist approach to teach users how to code (Peppler & Kafai, 2005), which focuses on learners constructing their own understanding and knowledge and is an approach similar to the Montessori philosophy of learning, I was curious to find out whether Scratch could be qualified as a Montessori material for students to learn programming skills. Therefore, I proposed an inquiry that would explore the following questions:

- How does Scratch help students develop digital literacy?
- Could Scratch be considered a Montessori material in a prepared environment?

Curriculum Analysis

Overview of Curriculum Theories

Curriculum theories have gone through a revolving door of change, with perspectives ranging from scholar academic to social efficiency to social reconstruction to learner-centred ideologies. Furthermore, each of these perspectives describe curriculum in its own way. For example, Marsh and Willis (2007) defined curriculum as "such permanent subjects as grammar, reading, logic, rhetoric, mathematics and the greatest books of the Western world that best embody essential knowledge" (p. 9). They view it as a syllabus or a structure of a discipline, and they view teaching and learning as the transmitting of a body of knowledge from teacher to students: students are empty vessels, and teachers are there to fill them with knowledge. This scholar academic perspective of curriculum differs from Bobbitt's (1918) version of curriculum as "that series of things which children and youth must do and experience by way of developing abilities to do things well that make up the affairs of adult life; and to be in all respects what adults should be" (p. 42). In his case, "what adults should do" are determined by cultural norms of society and the objectives should be developed and achieved in a way that promotes social efficiency. Despite the differences in definition of and perspective on curriculum, both assumed that the structure of education is from the top-down where an authoritative body governs what is to be learned.

Contrary to these views, social constructionist and learner-centred curricula shift the focus of education away from teachers and governing authorities and place it more on students by encouraging them to be active participants in learning. For instance, Freire (1970) summarized "authentic education" as something that "is not carried on by 'A' *for* 'B' or by 'A' *about* 'B,' but rather by 'A' *with* 'B,' mediated by the world" (p. 93). He posited that dialogues between teachers and students are essential, and that "without dialogue there is no communication, and without communication there can be no true education" (pp. 92-93). Not only that, these dialogues “cannot be reduced to the act of one person ‘depositing’ ideas in another, nor can it become a simple exchange of ideas to be ‘consumed’ by the discussants” (p. 89). Meanwhile, A. S. Lillard (2005) argued that children’s education should not be dictated by the delivery of a pre-established curriculum on a set schedule; rather, it is the children’s studies that “radiate from a core of deep interests into all curricular areas” (p. 326). It is important to note that this is not to say students are the sole determinants of the contents of the curriculum, but students should be put into consideration during curriculum design as well as during curriculum delivery. Student autonomy is valued if it is within the limits of what is constructive for the child and society as well as being able to get to the parts of the curriculum (A. S. Lillard, 2005).

The Montessori Philosophy

Maria Montessori (1870-1952) was an Italian psychiatric physician who was first known for her work with developmentally challenged children and found success when some them could achieve the same level of intellectual success as other children in a public school. With this, she began to devote her energy to work with children as an educator, opening her first day-care centre, called “The Children’s House,” in the slums of Rome to educate young children whose illiterate parents were too busy working to care for them. Taking the same methodologies and materials as the ones she used with the children with developmental disabilities, she soon discovered a child’s inherent ability to work and learn independently (P. P. Lillard, 1972). Montessori also observed the prospect of self-regulation and self-governance when she saw that while children were using the pieces of her apparatus, they tended to repeat the exercises until they were satisfied and put the equipment away. Also, one day coming into daycare realizing she had forgotten to lock away the materials for her children the night before, she was surprised to see that the children had already started working on the materials independently without the prompting of teachers. It is then that Montessori began to theorize on students' ability of independent learning and develop her education philosophy known as the Montessori Method that became the central tenet of Montessori schools around the world.

The Montessori Method is grounded on Maria Montessori's theory that there are four planes to human development from birth to adulthood, and learning is embedded in the experiences a student receives during each of the given planes. In order for a student to achieve maximum potential, learning experiences must be designed appropriately according to his/her progress and interests within a particular plane (P. P. Lillard, 1996). For example, In the primary years (from birth to age six), children learn about practical life skills not only to take care of themselves, but also to become independent learners who can make educated choices. During the next stage (from age six to twelve), students further develop self-direction through prepared environments (such as structured work cycles) and student-centric activities (such as inquiry-based learning that integrate subjects into topics) that follow individual student's desires to learn in particular areas within the framework of the curriculum. Throughout both stages, Montessori teachers do not act as lecturers but rather as facilitators of learning; their students are freed of adult interventions that help build resilience and independence.

One might think that Montessori education follows a learner-centred ideology, but there are traces of other curriculum ideologies embedded in Montessori philosophy and practice. For instance, one of the key components of the Montessori curriculum is Cosmic Education, which is a series of stories called the five Great Lessons that expose students to various academic disciplines such as Math, Language, Geography, History, and Science at a young age. Through the study of different timelines (e.g., Timeline of Humans, Timeline of Mathematics, and Timeline of Language) and dialectic discussions, students learn about the big picture of the world. The exploration of these stories "lead students to contemplate not only the past, but the future" so that "students are motivated to further research and work in the Montessori classroom" (North America Montessori Center, 2009). Unlike other activities in the Montessori classroom that are freely chosen by students, the Great Lessons are exposed to all students. In a way, the knowledge is introduced to students in a very scholar academic way.

Within a Montessori classroom, students are provided a variety of didactic, sensory materials that allow them to explore concepts from concrete to abstract. In order to keep track of student progress and determine when students are ready to move onto subsequent concepts, many Montessori teachers use programs that employ a levelling system. An example is the Albanesi Math Lab program. Before a student begins, he/she takes a pre-test that determines the level at which he/she starts at, and as the student reaches a milestone, he/she will write a post-test that determine whether understanding of the concepts covered is demonstrated. Even though students work through these programs at their own pace, they are moving along a set curriculum and follows an assembly-line-like system that resembles the social efficiency ideology.

Lastly, there are two reasons why I believe Maria Montessori is a social constructionist. First, when she first began her school, her intention was to educate children with developmental difficulties to acquire life skills so that they could perform practical tasks independently. She also hoped to bring education to children who were raised in lower-class families as their parents were too busy working to tend to them. It is easy to see the Montessori's vision of education for all so that her students can function independently regardless of ability and class is a Social

Reconstructionist belief. Moreover, the Cosmic Education that was mentioned earlier actually serves a secondary purpose:

> In the twentieth century, Maria Montessori saw Cosmic Education as a way to protect humankind from the threat of self-annihilation posed by seemingly endless acts of war and political aggression. In the twenty-first century, global awareness, cooperation, and peaceful communication, are integral to resolving problems Montessori could hardly have anticipated. Cosmic Education helps prepare children to deal successfully with today's realities. (Beydler, 2009, para. 1)

This view on education encourages students to investigate the world to communicate and act for the sake of peace and save the world from a certain destruction resonates with some of the Social Reconstructionist views.

All in all, my original thoughts on Montessori education as purely learner-centred changed over the course of my inquiry by examining curriculum through a critical lens. However, the underlying philosophy of Maria Montessori is still heavily aligned with a learner-centred ideology. At the same time, I realize that my own philosophy on teaching and learning, like Montessori's, does not purely align with a particular ideology and is influenced by all of the aforementioned ones. In fact, I was inspired to develop my own pedagogical philosophy of curriculum that guides my teaching practice.

My Personal Definition of Curriculum

Although there are many different definitions of curriculum that currently exist in academia, none of them matches my education philosophy that is partly influenced by my belief in Maria Montessori's philosophy. Teitelbaum (2008) briefly mentioned that "some curricular scholars posit curriculum as the interrelated complex set of plans and experiences that students have under the guidance of the school" (p. 171), and I have constructed my own personal definition using it as a basis. In my view, I think curriculum should be the interrelated complex set of plans and experiences that *individual* students have under the guidance of *teachers that foster the development of the students to be independent learners in life.* This definition shares Teitelbaum's underlying assumption that the teacher is the passive participant in learning, and that what a student undertakes is emphasized. Moreover, it assumes that learning should be lifelong. As a person develops, they continue to make connections with the ever-changing world. It does not stop until they stop developing intellectually at the end of their life.

Under my definition, the primary role of teachers is to establish a prepared environment in which students can develop their intellectual skills internally. In this setting, knowledge is meant to be actively discovered by the students depending on individual interests. It is not something that is rote memorized but to make connections and understand where the answer comes from through their senses. The teacher needs to encourage intrinsic motivation for learning from students by letting them self-correct their mistakes. Also, learning should be a social process in which collaboration is encouraged so that, from one another, students can construct their own understanding of the world around them.

The purpose of learning, and the aim of the curriculum, is to allow students to grow and become independent members of society. It is through independence that a person can achieve the success set out through their own agenda.

Montessori Education and British Columbia's New Curriculum

Over the past few years, educators in British Columbia worked to develop a new curriculum for students in the province that began its implementation in September 2016. The redesigned curriculum features aspects that put an increased emphasis on the learners in the education system compared to the previous versions as the Ministry of Education attempts to move towards learner-centred school environments that are more flexible. The Ministry of Education stresses that "learning can take place anywhere, not just in the classrooms", that "although the learning standards are described within areas of learning, there is no requirement for teachers to organize classrooms, school or instruction in this manner", and that "teachers are encouraged to create courses, modules, thematic units or learning experiences that go beyond learning area borders to focus on students' needs and interests or local contexts" (BC Ministry of Education, 2015b, Flexible learning environments section). In the context of learned-centred ideology, this allows for implicit curriculum to be valued in a greater extent in which students are able to explore areas of interest that are not explicitly documented in the curriculum. These statements also resonate with the Montessori's philosophy in that teachers are given the flexibility to work with students to come up with topics to be learned.

There are other facets in the new curriculum that echoes with the learner-centred approach to learning. When compared to the current, older version of the curricular documentation, the government puts more acknowledgments on the fact that not all students learn successfully at the same rate, in the same environment, and in the same ways. It also realizes that interests motivate student learning. Thus, the Ministry of Education hopes that the new curriculum "places learners at the centre of the learning landscape, and encourages motivation, curiosity, and active engagement" (BC Ministry of Education, 2015c, Renewed Provincial Curriculum section). To do so, students are given more choices by having classrooms employ an inquiry-based approach within the draft curriculum in each subject area. Not only are potential questions relating to the prescribed contents suggested to teachers, students are also encouraged to form their own questions to explore and branch out from the topics discussed in the classroom. This places ownership of learning back to the students, and they are ultimately responsible for their own learning. Moreover, by recognizing that individual students are unique, the new curriculum hopes to “encourage students to be co-operative, principled, and respectful of others regardless of differences” (BC Ministry of Education, 2015a). In a way, this is similar to Montessori and A. S. Neill’s arguments that freedom is not licensed: while students are granted freedom to explore topics pertaining to their interests, they must not interfere with other students’ freedom as well. They must respect each other’s differences in interests, needs, and learning style and be constructive for oneself and society (A. S. Lillard, 2005; Readhead, 2008).

Perhaps the biggest direction that the Ministry of Education took towards creating a learner-centred environment is the establishment of the core competencies, which places more emphasis on skills that a student will need to be successful in life and shifts some of the focus on subject-specific outcomes. Since core competencies can be observed over a long period of time — from kindergarten to grade 12 — different students can be at different points or “profiles,” and teachers can use these observations to tailor instructions and activities to individual students to promote their growth in those areas. These profiles are created based on examples of student-led activities rather than teacher-led ones, and are written from a student’s point of view using “I” statements. They are also progressive in nature and could be viewed as analogous of the planes/stages of development suggested by Maria Montessori (1949) and Piaget (1952).

The core competencies are divided into three major areas: thinking, communication, and personal and social. Each of these competencies promote personal relevance in student learning.

For instance, the thinking competency encourages students to be critical and creative thinkers who can think for themselves as well as using existing information to create new meaning that is relevant to them. The communication competency aims to provide students with a voice to speak and discuss with teachers about their needs and interests in order to fulfill their educational experience, and the personal and social competency wants students to recognize that each of them is unique and that while there is freedom in what they choose to do, they must realize that it is not licensed without regards to people around them. Unsurprisingly, these core competencies also align with Maria Montessori's philosophy (Montessori, 1912, 1917, 1949).

The New Applied Design, Skills, and Technologies Curriculum

Today's generation of students, coined "Digital Natives", are living in a ubiquitous environment of technology where tools are becoming more common and more easily accessible (Prensky, 2001). Furthermore, over the past two decades, education has gone through a shift in focus towards having students acquire the skills to allow them to be successful in 21st century society where information is created, accumulated, and accessed at an unprecedented rate. Therefore, it is important to recognize the importance of equipping all students in BC with these foundational skills while developing their capacity to harness the power of digital technologies. Not only would they be able to apply the skills in cross-curricular learning settings, they would also be better prepared to explore and select specializations to follow their passion as they enter the work force. As part of the new curriculum, the Ministry of Education introduced the addition of the Applied Design, Skills and Technology (ADST) curriculum to be implemented beginning September 2016, in which digital literacy became one of the key components in the middle school level. Beforehand, computer programming was a concept that was only taught in grades 9 through 12 (BC Ministry of Education, 1996, 2003). With the expansion of computational thinking into the grades 6 to 8 level in the ADST curriculum, middle school students will be exposed to coding at an earlier age. As discussed in my literature review in the next section, there are discourses on the definitions of digital literacy and how computer programming plays a prominent role in the development of digital literacy in students.

Our world is undergoing constant change, and so is education. To help students in British Columbia become successful in the 21st century, the Ministry of Education developed a new curriculum that is more learner-centred and puts more emphasis on skills over content. This approach aligns closely with Maria Montessori's philosophy on learning. However, while Maria Montessori seems to be ahead of her times by promoting some of the 21st century skills such as critical thinking, collaboration, character, and culture and ethical citizenship in her curriculum, an area that is rarely mentioned — likely due to the unforeseen advancement of technology — is competencies around computer and digital technologies. With the introduction of the technology-focused ADST curriculum that is part of the new curriculum, it has become a rather unchartered territory for Montessori educators who incorporate little digital technology use in their classrooms. Therefore, through an extensive literature review and a case study of teaching students programming skills using Scratch in the following sections, I conducted the study in order to connect the Montessori pedagogy with British Columbia's new curriculum.

Literature Review

My inquiry was broken into two parts: how Scratch could help develop digital literacy, and whether Scratch could be qualified as a Montessori material to do so. As a result, my review of literature ranged from theories surrounding the philosophy of Maria Montessori, to the conceptual frameworks and reports on digital literacy as part of 21st century learning, to both quantitative and qualitative research on the benefits of learning computer programming skills.

The literature I reviewed was in a variety of formats. Most of the literature was peer-reviewed articles from academic journals as well as professional magazines contributed by reputable figures from academia, such as teachers, professors, and educational psychologists. Some of the findings were taken from masters or doctoral dissertations and conference proceedings. Information regarding definitions of digital literacy was extracted from reports generated by government and non-profit organizations, such as MediaSmarts and the International Society for Technology in Education (ISTE). Lastly, several books and online resources (e.g., webpages) on the Montessori philosophy were used. Table 1 in the Appendix summarizes the number of resources reviewed for each type of publication.

Most of these resources were obtained from academic databases, such as JSTOR and ERIC, through the University of British Columbia. Some research papers were made available on ResearchGate, while the rest surfaced during searches on Google. It was important that the information searched was not too wide nor too narrow for the purposes of the research; the queries employed include the following terms, or a combination of:

- “Digital literacy”
- (Computer *OR* Technology) *AND* Montessori
- “Montessori materials” *AND* Characteristics
- Computer *AND* (Programming *OR* Coding)

Selection Criteria

An extensive search using the aforementioned query term yielded 65 sources of various types, and a summary can be found in Table 2 in Appendix A. In order to further narrow down the number of works and focus on my inquiry, the two sub-questions that I established were broken down and organized into five areas of debate for investigation.

First, with an increasing focus on developing 21st century skills in the classroom (C21 Canada, 2012; P21, 2015), there were claims that these skills have already been embedded in 20th century Montessori pedagogy (Powell, 2009; Torrence, 2012). Secondly, the promoting of digital literacy in Montessori classrooms has been slow, as there have been opposing views on the incorporation of computer technology in Montessori learning environments. Third, I examined different discussions on the characteristics of Montessori materials and established assessment criteria for what could be considered a Montessori material. These criteria would then be used to evaluate Scratch, a programming language that I believe could help develop digital literacy in a Montessori classroom.

Subsequently, to better understand whether Scratch could help develop digital literacy, I looked into different literature on the definitions of “digital literacy” since there are a multitude of perceptions around the topic (Lankshear & Knobel, 2008) and it was important for me to establish one on which I could ground my claim. Lastly, an overview of computer programming and its connections to developing digital literacy was focused upon for discussion.

As a result, a portion of the literature gathered was not considered for the study. Of these, five looked into the Montessori philosophy but did not have concrete connections to 21st century learning. Four articles provided quantitative research that falls outside the K-12 classroom setting, which is the scope of my inquiry. One research paper focused its study on using technology in a Montessori secondary setting, though no connections between digital literacy and Montessori philosophy were established in the author's claim as her primary goal was to look at the effects of integrating mobile learning on high school students. Two resources were already parts of other reviews that I included, and one merely paraphrased from one of Maria Montessori's books that I will also discuss. Lastly, there were articles referenced in some of the literature that could not be accessed but could provide further reading.

Discoveries

Montessori and 21st century learning. With the rapid advancement of digital technology that resulted in the unprecedented rate at which information could be created, accumulated, and accessed, governments around the world have been promoting the shift in pedagogy from a social efficiency perspective to one that takes on a learner-centred approach and focuses on skills that deal with information (C21 Canada, 2012; P21, 2015). National non-profit advocates such as C21 Canada and the Partnership for 21st Century Learning (P21) have created frameworks to outline the skills students need to adapt to this new-age society. Since C21 Canada developed its framework with research on other frameworks including P21 and my research will be done in Canada, I have decided to adopt the competencies outlined by its report.

C21 Canada (2012) has established seven 21st century competencies: character, collaboration, communication, computer and digital technologies, creativity, critical thinking, and culture and ethical citizenship. Powell (2009) and Torrence (2012) claimed Maria Montessori was ahead of her times, in the sense that her pedagogy has already embraced some of these 21st century skills in classrooms guided by her 20th century pedagogy. In fact, a closer examination of other literature on Montessori's philosophy will reveal that these competencies

— with the exception of computer and digital technologies — have been regularly promoted in Montessori classrooms. A summary can be found in Table 3 of Appendix A.

A central tenet behind the Montessori philosophy is the guiding of children towards independence (Montessori, 1912). As an educator who first worked with children with developmental disabilities, Dr. Montessori discovered that children have the inherent ability to work and learn independently in a carefully prepared environment (P. P. Lillard, 1972). In that environment, the teacher should act as a guide on the side to elicit and follow the children's interests and intrinsic motivation so that education could reach its goal of leading the children to lifelong learning (Adams, 1970; P. P. Lillard, 1996; Montessori, 1949; Torrence, 2012). The idea of self-directed and lifelong learning matches the *character* competency prescribed by C21.

Montessori (1912) also stressed that her program "involves much direct social experience, both in the general life of the school and in the manual work done by the pupils" (p. xxiii). Powell (2009) and Torrence (2012), faculty members of the Centre for Montessori Teacher Education and former president of the American Montessori Society (AMS) Board of Director, respectively, described Montessori's attitude of students being placed in three-year multiage groups so that *collaborative* learning and personal connections can be supported. In this environment, the youngest (awestruck follower), the middle child (observant apprentice), and the eldest (experienced and nurturing leader) work together to enrich learning experiences.

The competency of *communication* is crucial to collaborative learning and Maria Montessori believes that the key to communicating effectively is through the learning of languages. Indeed, as Helfrich (2011) wrote in her guide to Montessori learning in the 21st century, "the impact of learning a language as the key to the development of communication skills is tremendous" (p. 130). Dr. Montessori developed many materials, such as grammar parsing symbols, sandpaper letters, and sentence analysis boxes that could help aid a child "make this difficult conquest of language so that he can enter into communication with other beings" (Montessori, 1949, p. 134).

Dr. Montessori's vision of education also encompassed *creativity* and *critical thinking* competencies; although her books did not directly reference these skills, she viewed imagination as a special gift of nature (Montessori, 1917, p. 115), and that her materials are "intended primarily for creative use by the children and offer opportunity for mathematical analysis and the teaching of design; and its procedure is rich in resources for imagination" (Montessori, 1912, p. xxv). Many individuals (Cooney & Jones, 2011; Cossentino & Brown, 2015; A. S. Lillard, 2013) who studied or were inspired by her work could vouch for her values. For instance, Cossentino and Brown (2015) argued that "fully implemented Montessori learning environments will yield high performance on measures of executive function and creativity", and that Montessori methods could be used as an intervention to nurture creative and critical thinking in the public school system. Since their study is still in progress, their findings could provide more insight in these areas in the future.

Finally, Dr. Montessori is an advocate of peace (1912, p. 347; 1949, p. 407). Her teaching of grace, courtesy, and conflict resolution skills to children at a very young age prepares them to be caring and peaceful citizens in the world (Torrence, 2012). Her idea of a Cosmic Education where students are taught that everything, including humans, is interconnected provides the underlying rock on which the concepts of harmony and peace are built. These concepts also form the basis for the *cultural and ethical citizenship* 21st century competency.

Technology and Montessori Education. As one can see, the literature I reviewed demonstrated how the Montessori pedagogy has long been promoting all of the 21st century competencies in classrooms, with the exception of computer and digital technologies competency (Torrence, 2012). In fact, Tosco (2015) argued that the Montessori environment in particular rarely incorporates technology in any consistent way. One reason for this could be that this area is relatively new and that "there are limited peer reviewed studies or researches on the effects of integrating technology into elementary Montessori classrooms" (p. 9). However, a bigger reason could be the fact that there are opposing stances in regards to the attitude towards incorporating computer and digital technologies (referred as technologies from this point on) in the classroom. Seventeen sources around this controversy have been selected for review. Of these, nine papers fully supported the integration of technology in the Montessori learning environment, while three have mostly negative comments about it. The remaining five took on a more neutral stance and provided an analysis of both pros and cons of adopting the use of technology in Montessori education. Furthermore, several researchers made recommendations of how to incorporate technology into a Montessori classroom.

On the proponents' side, they argued in their work that there are already intersections between Montessori education and digital learning (Dunn, 2000; Hubbell, 2003; Schneider, 2012). In her study to see if increased access to and use of technology have a positive impact on her students' learning through activities in writing, Dunn (2000) concluded that technology could indeed help her students write well by allowing for creative and collaborative work in her classroom. One criticism though, is that she is only using writing in her case study, which may be difficult to extrapolate that technology is beneficial to the overall learning experience in a Montessori classroom. Hubbell (2003) explained how technology has made a seamless integration into her Montessori classroom because the computer has helped her students in completing authentic tasks that would otherwise be impractical or inefficient to do so without it. One drawback of the research was the lack of findings on the limitations from such integration. Schneider (2012) on her blog post discussed how technology could provide online and blended environment that follows individual learning progression and eliminates age and grade restrictions. She believes that online learning can provide formative feedback in real time, that students are active participants in self-guided learning, and the removal of geographic barriers allow for learners to connect world-wide and establish a global citizen perspective. Her arguments were valid, though the fact that she did not reference any empirical studies and her work was published in an informal platform makes the findings less persuasive than others.

Dr. Montessori once said, “What purpose would education serve in our days unless it helped [hu]man[s] to a knowledge of the environment to which he [or they] have to adapt” (1949/2007, p. 11)! Her ways of teaching always promote practical life skills that children can find useful in order to lead independent lives. For children in this generation, being literate in technology is a practical life skill that will equip children with the tools to create, explore, and share (Chattin-McNichols, 1996; Hubbell, 2003; Powell, 2015; Tosco, 2015). Dr. Montessori also stressed the importance of following the child’s needs and “encourage the flowering of a child’s natural gifts” (as cited in Schneider, 2012). Given that classrooms should be reflections of home, community, and the world, technology can provide the tools to engage the digital natives by extending their curiosity into the digital air of the 21st century (Tosco, 2015; Woolsey & Woolsey, 2008).

Despite strong support given by the proponents, opponents who argued against the integration of technology also referenced Montessori’s philosophy in their cases. More specifically, they felt that technology goes against the nature of using Montessori materials and that it could stunt educational growth (Berger, 2016; Goodwin, 2010; A. S. Lillard, 2005). Goodwin (2010) quoted Maria Montessori, saying "[n]ever give more to the mind than you give to the hand" (p. 8). He argued that technologies like the iPad provide virtual objects that are only representations of reality and do not support true hands-on learning that comes with manipulating a physical, three-dimensional material. However, one could argue that learning certain abstract concepts such as programming is not achievable without a virtual environment. A.S. Lillard, a well-established Montessori researcher, claimed that "Montessori environments offer children that [quietness]; most computer software for children does not" and "many computer programs regulate children's attention for them, rather than helping them learn to regulate their own attention" (2005, p. 336). It should be noted that the quantifiers "most" and "many" were used and without giving specific examples, her statement is not a generic statement that can generate strong persuasiveness. Lastly, Berger (2016) alleged in his article that games on digital devices engage children with passive activities and instant gratification that lead to the deterioration of emotion regulation, inhibitory control, and control with the real world. The fact that he inferred all activities provided by technology as games is a major fallacy in his argument.

There are other researchers that provide a more neutral view on integrating technology in the Montessori environment. The views that they shared echoed the potentials and limitations that were pointed out by the proponents and opponents mentioned above (Buckleitner, 2010; Herman, 2012; Montminy, 1999: Prozesky & Cifuentes, 2014). Love and Sikorski (2000) conducted a qualitative study and provided feedback, both positive and negative, from eleven Montessori educators on technology use in the classroom. They also provided some general recommendations for using technologies without compromising the inherent Montessori philosophy. For one, the use of technology must be observed and studied to maximize the potential and minimize downsides (Buckleitner, 2010). Herman (2012) strongly encouraged educators to "understand the consequences of excess interaction with technology on children and help them construct the ability to balance the instant gratification of technology with the mind's deeper engagement in tasks and mental substance" (p. 36). Some, like Montminy (1999), suggested a set of extensive criteria that technology tools could follow in order to reap the benefits while following true to Montessori education.

Even literature that fully supports technology integration provided recommendations for integrating technology in a Montessori classroom. For example, technology needs to promote authenticity (doing work that matters), autonomy (freedom to choose and control over own activities), connection (interaction with both adults and peers), and inquiry (problems or questions posed in a way that is compelling or relevant to children) (Hubbell, 2006; Powell, 2015). Also, computer programs should be open-ended and explorative in order to expand children's creativity and help them "synthesize what they have learned and relate it to other knowledge" (Chattin-McNichols, 1996, p. 38).

The most important recommendation of all, as mentioned in a number of reviewed works, is that technology should not be the be-all and end-all of education. It should be viewed as a means towards an end, which in Montessori sense, is to prepare children to become independent, life-long, and peaceful learners and useful, contributing members to society (A. S. Lillard, 2005, p. 335). Technology is complementary in the sense that it could be used as an assistive technology (Boyd, 2008) to extend Montessori experience, to perform tasks that otherwise could otherwise be impossible or unfeasible (Chattin-McNichols, 1996), all the while not replacing traditional methods and time-tested materials and activities and "[redirecting] internal development or [interfering] with our students' intrinsic understanding of the value of work" (Dunn, 2000, p. 35).

Overall, despite some negative views on integrating technology in Montessori education, many believed that as a visionary who revolutionized education in the 20th century, Maria Montessori would not only approve but also embrace technology in her classroom (Love & Sikorski, 2000; Prozesky & Cifuentes, 2014). As both Powell (2015) and Hubbell (2006) advised, technology and digital experiences can open the door to authenticity to students as well as give the world to the child by bringing her or his classroom experience in line with his experience of the world outside.

Montessori materials. In Montessori education, there are specific materials that are part of the learning environment which students can physically manipulate in order to grasp concrete concepts. It is only through these concrete experiences that children construct or extrapolate abstract knowledge that ties to the real world (Dennis, 1974, p. 6). These traditional materials were originally derived from didactic materials developed by Seguin, from whom Montessori received her inspiration (Hardy, 1917). There are five types of materials in general: (a) cultural and artistic, to aid the child in self-expression and learn about how things are interconnected through biology, geography, and history; (b) literacy, used to help the child recognize the makeup of a language in order to communicate; (c) mathematical, to present abstract concepts in concrete forms; (d) practical life, whose aim is to help the child transition smoothly between home and school and usually consist of household items; (e) sensorial, which appeals to the child's senses and to stimulate them in order to improve sensory perception from coarse to fine (Sihirli Bahçe Montessori School, n.d.).

In her book, Paula Lillard (1972) explained that Montessori materials exhibit five underlying principles. First, concepts are isolated and presented to the child one at a time. This way, it “simplifies the child’s task for him and enables him to perceive the problem more readily” (p. 61). Materials are also progressive from simple to complex, and from concrete to abstract. When concrete knowledge has been instilled into the child, “a new physical picture, a species of higher plane in the complex development, is revealed…and his mind is capable of abstraction” (p. 63). Moreover, materials are designed to prepare the child indirectly for future learning. This way, the child is able to “experience success in his endeavors much more readily and aids the development of self-confidence and initiative” (p. 62). Lastly, Montessori materials must provide a control of error, so that not only is the child is permitted to make mistakes, he or she can also recognize their mistakes and learn from them on their own without adult guidance.

Lillard (1972) also stressed that these materials should only be presented to the child at the right moment, that is, when the child has mastered the previous concept and is ready for the next level of abstraction. This aligns with Montessori’s philosophy of following the child’s needs, and is the only way to capture her or his attention and allow them to maintain concentration (p. 60), a state of mind that Montessori defined as “normalized” (Montessori, 1949, p. 302). Finally, Montessori materials must not only engage the child’s hands, but also her or his mind as well. Together, he or she is an active learner and is not passively waiting to receive instructions from the teacher.

One of the goals for my inquiry is to determine whether Scratch could be considered as a new Montessori material. Although Lillard (1972) presented a very good set of underlying Montessori principles on which materials should be based, Prozesky and Cifuentes (2014) established an expanded list of ten principles that technology tools can follow in order for them to be integrated as part of the prepared environment. These principles are: planes of development, engagement of the senses, learner concentration, isolation of concept, control of error, concrete-to-abstract sequencing, creativity and imagination, intrinsic motivation, collaboration, and peace. Referencing a comprehensive list of works, including Montessori's work and papers that were also included in this review, they concluded that "Montessori's philosophy aligns precisely with theories of instructional design and education technology" (p. 29) and "technologies such as those created by Montessori facilitate active, tactile experiences that develop proficiencies in children as described by her ten principles" (p. 36). This set of criteria was used as an evaluative tool for Scratch in my findings analysis.

Even if a tool meets all of these criteria, Angeline Lillard (2008) sternly cautioned educators when introducing new materials into the Montessori environment and that the same degree of considerations that Maria Montessori did in coming up with the traditional set should be made. When developing materials, Maria Montessori made sure that the materials suited the needs of her students by watching her students use them and revising and refining them until they meet the specific needs. As well, while a material serves many purposes, there is minimal redundancy across all of materials; any redundancy should have a well-aimed intention. Montessori also ensured that her materials not only work together within the classroom, but they are versatile enough to work across classroom levels while fitting into the proper sequencing of skills. Lastly, educators need to be aware of the novelty effect on students; that is, new materials tend to attract more attention from students thus leaving traditional ones to gather dust in the corners. It is important that students regularly be reminded that traditional Montessori materials are very special. These considerations were taken into account when proposing that Scratch be integrated as a new material in a Montessori classroom.

Digital Literacy. In order to investigate whether Scratch could help students develop digital literacy, it is important to establish a clear definition of digital literacy to ground my investigation. The idea of digital literacy has undergone two major shifts over the past four decades. Kafai, Peppler, Lemke, and Warschauer (2011) explained that during the 1980s and 1990s with the introduction of Seymour Papert's LOGO programming language, programming was seen as a new form of creative media production and gained enough popularity to reach "near-universal presence." However, with the introduction of the World Wide Web and with the idea of navigating multimedia applications gaining traction, learning how to program was placed on the back burner as, albeit exaggerated, "searching the Internet and preparing PowerPoint presentations became the hallmark of digital literacy in school settings" (Kafai et al., 2011, p. 94).

With the Web moving to version 2.0 at the onset of the new millennium, the idea of digital literacy has once again changed and an emerging concept was adopted to derive a new measure of the quality of learners' work in digital environments (Eshet-Alkalai, 2004). No longer about the ability to use a piece of software or operate a digital device, digital literacy became "a multi-faceted skill that covers the ability to find, use, interpret, modify, and create a variety of digital media" (Spector, 2013, p. 22). Chase and Laufenberg (2011) offered a simplified interpretation by focusing literally on literacy, that digital literacy is reading and writing across multiple modalities. On the other hand, Eshet-Alkalai (2004) even went as far to include photo-visual literacy, reproduction literacy, branching literacy, information literacy, and social-emotional literacy.

As can be seen, there has not been a standardized definition for digital literacy that is agreed upon. In fact, in their book *Digital Literacies: Concepts, Policies and Practices*, Lankshear and Knobel (2008) proposed pluralizing literacy to literacies due to “the sheer diversity of specific accounts of ‘digital literacy’ that exist, and consequent implications of that for digital literacy polices” (p. 2). Because of this, rather than choosing one of these definitions from individual scholars or merging existing ones to form my own, I looked to ones defined by more established organizations like C21 Canada, P21, and MediaSmarts. These are non-profit, nation-wide organizations that continually do research in the field of education in the hopes of informing the public. The following section provides an overview of their position on digital literacy.

In C21 Canada’s report *Shifting Minds 2.0* (2012), the term digital literacy was mentioned as part of the computer and digital technologies competency in its 21st century skills framework, which is defined as “[t]he capacity to use computers and digital resources to access information and create knowledge, solutions, products and services” and “[t]he capacity to use social media for learning” (p. 12). In this definition, the term “capacity” indicates that there could be a spectrum of student ability, which is a positive since there’s no inherent system of ranking student achievements. However, C21 Canada’s wording places an emphasis on using information and communication technology (ICT) tools over the actions of accessing information or creating content. Moreover, the report did not provide an assessment model to accompany its framework.

P21 (2015) does not use the term digital literacy but refers to it using “ICT literacy” instead. The organization also defined ICT literacy as a result of combining various learning skills such as thinking and problem solving skills, information and communication skills, and interpersonal and self-direction skills with the use of 21^{st} century technologies that are referred to as tools (p. 11). Under this conceptual framework, ICT/digital literacy is given a more holistic view that encompasses all of the other competencies featured in C21 Canada's report. However, viewing ICT/digital literacy from this lens lends itself to the question of whether it is what 21^{st} century skill is about; that ICT/digital literacy is an end but not a mean. This could be disputable by people who refuse to look at ICT as the key piece to revolutionizing education. In addition, like C21 Canada, P21 focuses students' ability to use 21^{st} century tools to perform learning skills. P21, also similar to its Canadian counterpart, did not provide any assessment strategies, despite suggesting the use of formative assessments to evaluate student proficiency in this area.

A third organization, MediaSmarts (formerly Media Awareness Network), provided an "established and internationally accepted definitions of digital literacy" based on three principles (Media Awareness Network, 2010, p. i):

- the skills and knowledge to access and use a variety of digital media software applications and hardware devices, such as a computer, a mobile phone, and Internet technology
- the ability to critically understand digital media content and applications
- the knowledge and capacity to create with digital technology

This definition, like the previous two, values the ability to access and use technology. However, it also establishes an equal importance for understanding and creating digital media. Furthermore, unlike C21 Canada and P21, it provides a recommendation for assessing students using the standards and performing indicators developed by ISTE. As a result, this was the definition I adopted for my inquiry.

Programming. In my opinion, computer programming (or coding) is an activity in which humans establish a set of instructions using a language that is understandable and meaningful by a computer in order to perform a task or solve a problem. With the shift to Web 2.0, there has been an increasing emphasis on the participation in and production of media (O'Reilly, 2005), and media production lends itself to the need to learn programming skills (Kahn & Spiegel, 1999). Navarette (2013) pointed out that programming is central to technology literacy, just as writing is central to traditional text-based literacy. Indeed, there are many literature works out there to support his point, and I have decided to use ISTE's *Standards for Students* (2016) to review the literature. Newly revised, ISTE provides seven profiles of achievement indicators to measure students' digital literacy in the format of roles that students can assume. These profiles are: Empowered Learner, Digital Citizen, Knowledge Constructor, Innovative Designer, Computational Thinker, Creative Communicator, and Global Collaborator.

Students who are Empowered Learners have the ability to use technology to actively choose, achieve, and demonstrate competency in their learning goals. In a qualitative study done by Kahn and Spiegel (1999), educators were interviewed on the role of computer programming in education. Their summary of findings indicated that there is a potential of programming to affect metacognitive and self-regulation skills that could be furthered researched on, and that whenever mistakes are made, feedback provides a learning experience for the programmer. Yang and Chang (2013) echoed this through their quantitative study of introducing seventh-grade students to participate in digital game authorship, from which the authors concluded, with empirical data, that an increase in concentration, critical thinking, and academic skills was observable. Meanwhile, Akcaoglu (2014) in another quantitative study, which featured an extensive literature review, found that deep thinking skills that empower learners could be trained through the process of debugging a piece of code.

Part of being digital literate is the ability to become Digital Citizens who can recognize rights, responsibilities, and opportunities in the digital world. It is important for students to learn to use their acquired coding skills responsibly and to not attempt to gain unauthorized access to systems and private information by hacking. When using programming to create digital media, learners also need to be aware of ethical standards as both media makers and participants in online communities (Peppler & Kafai, 2007).

As Knowledge Constructors, students construct knowledge and experiences by exploring real-world issues and problems and discovering and developing ideas in the pursuit of answers. When designing a system, a number of interrelated variables and parameters must come together in order for it to be complete and functioning. This often lends itself to the practice of problem-solving where the problems are ill-structured (Akcaoglu, 2014). Programmers need to understand these problems before figuring out solutions that would be built into the applications' functionalities, and this may involve using effective research strategies to locate accurate and relevant information to build a knowledge base.

Programming also challenges learners to become Innovative Designers by solving problems using new, useful, or imaginative solutions. Learning how to program entails the understanding the iterative process of design (Kahn & Spiegel, 1999) in which testing and refining often lead to the fostering of creativity in learners. Navarrette conducted a qualitative case study and a detailed literature review on creative thinking, technology, games and play, and game creation. He claimed that the game design learning approach enabled students' individual creative potential, provide a rich and enjoyable learning experience with authentic technology, and promote deep and insightful learning (Navarrete, 2013).

By design, programmers are Computational Thinkers. They are involved in problem-solving, algorithmic thinking, higher order thinking, concept abstraction, and understanding the proper vocabulary (Israel, Wherfel, Pearson, Shehab, & Tapia, 2015; Webb, 2013). By learning how to program, one has the opportunity to practice reframing seemingly difficult problems into ones that he or she knows how to solve and transfer this skill into everyday life (Wing, 2006). For instance, to attempt to recover a misplaced car key one would retrace his or her steps; to decide which cashier to lineup is to evaluate performance and efficiency. Therefore, one way to have students engaged in computational thinking is to engage them in programming a computer (Denner, 2011).

Unlike in the past when learning coding is about learning algorithms and data structures, coding nowadays is about making applications that are tangible and sharable with others (Kafai & Burke, 2013), which means programmers can become Creative Communicators who express themselves in a variety of ways for an intended audience. In traditional text-based literacy education, one of the major challenges is to teach young writers to anticipate the needs of the reader. In my experience working with middle school writers, many of them often omit details and logical explanations because they assume their readers would understand what they were thinking about. By having students program interactive media such as games for each other, they learn to better understand their audience because of the "added complexity for both player and designer, including the challenge that the player…can move around inside the world of text and experience it from more than one visual, spatial, and textual perspective" (Robertson, 2012, p.

386). In addition, part of good programming practice is the ability to write code that is concise, easy to follow, and well documented so that it is easier for another person to debug the code.

Last but not least, programming can encourage students to be Global Collaborators who work together to examine issues and problems from multiple viewpoints. As with any ill-defined problems, the focus on creativity when finding solutions is what makes computing so highly collaborative (Israel et al., 2015). Nowadays with the availability of online file-sharing tools, collaborative programming has never been easier, based on my experience as a computer scientist: Collaborators no longer need to sit in front of the same computer; they could be across the globe from one another and still elicit synchronous feedback and discussions. At the same time, code repositories feature versioning control systems that allow changes to be tracked and merged and allow work to be done asynchronously. With geographical barriers removed, it is easier and more intrinsically motivating for students to become contributors rather than just consumers of digital media.

To conclude, teaching students how to program is an effective way to help them develop digital literacy. Using ISTE's 2016 *Standards for Students* to examine and review a number of works, I have demonstrated coding activities allow students the opportunities to develop some or all of the skills that fall under the seven achievement indicators. A summary that outlines how these literatures reference the standards can be found in Table 4 in Appendix A.

Research Design

The study took place over twelve months, with data collected from two participating middle school classes (Grades 6-8, mixed gender, with a variety of digital skills). One class was composed of 18 participants in the Montessori program, while the other class included 26 participants in a neighbourhood program. **Error! Reference source not found.** below shows a demographic breakdown of the participants in my inquiry.

	Number of Males	Number of Females
Group 1 (Montessori Program)	11	7
Grade 6	6	4
Grade 7	3	2
Grade 8	2	1
Group 2 (Neighbourhood Program)	15	10
Grade 7	4	4
Grade 8	11	7

Table 1. *Demographical breakdown of participants in this study*

My research was broken into three main stages: Exploration, Data Collection, and Analysis (**Error! Reference source not found.**). The process was identical between the two groups of participants, but slight adjustments were made in the second iteration based on the limitations observed in the first iteration. The stages are detailed in the following sections. A comprehensive timeline of the research project could be found in Appendix B.

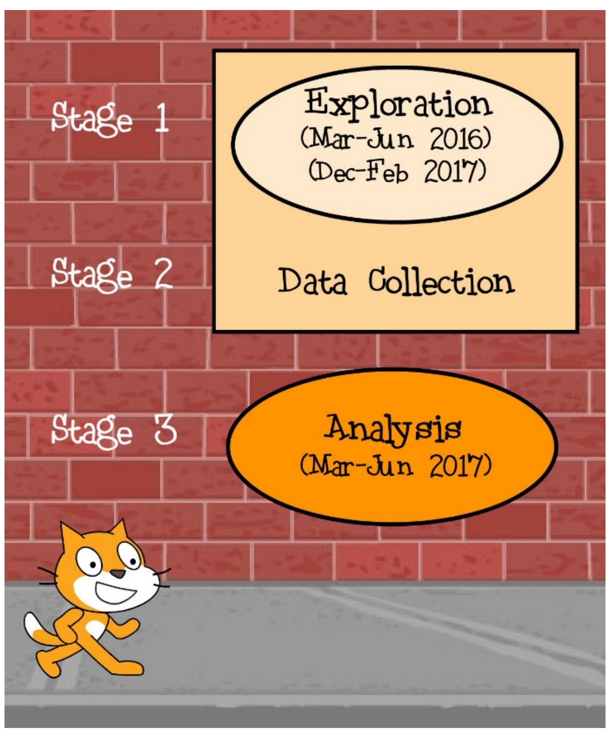


Figure 1. A framework of the stages of my research.

Stage 1: Exploration. The first stage of my research consisted of three sub-stages. In the first sub-stage (Discovery), I introduced students to Scratch by taking them to the Scratch website as well as showing them two videos of some of the projects that were produced by users around the world. Afterwards, they were given the opportunity to explore the website on their own and interact with various user-created projects. During this time, parent permission letters (Appendix E) were sent home. After collecting all the paperwork, I created accounts using pseudonyms and assigned them to each of my students.

After students received their accounts and were given a mini-lesson on how to log in and access the different sections of their accounts, we moved onto the second sub-stage (Learn). In this sub-stage, I asked students to complete self-guided tutorials that are provided by Scratch. These tutorials allowed students to learn how the various blocks of code work, and exposed them to different types of projects they could create using the program (**Error! Reference source not found.**). Within each tutorial, students followed step-by-step instructions to build sample projects. The instructions were mostly text-based but contained pictures to show what newly introduced blocks look like, as well as occasional short animated GIFs that demonstrated more complicated steps. As students finished each tutorial, they shared it onto their public Scratch profile so I could check-in on their progress.

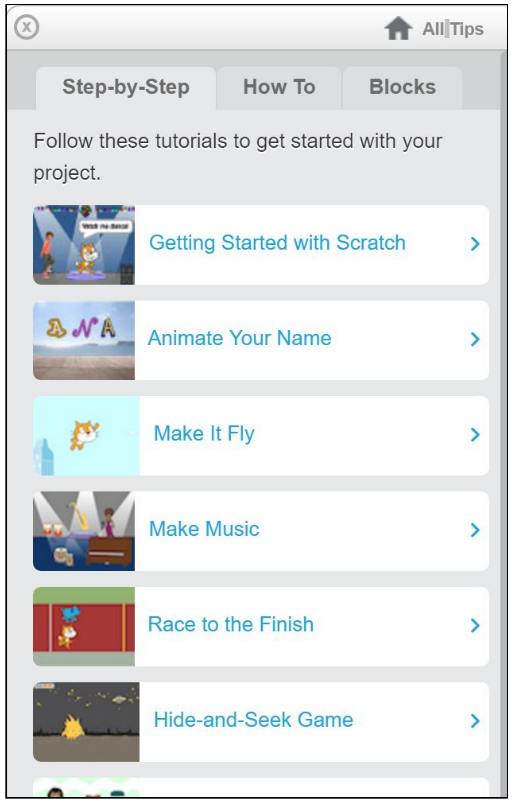


Figure 2. A sample list of tutorials available for students to complete in Scratch.

When students have completed a sufficient number of tutorials to become confident in coding independently in Scratch, they proceeded to the third sub-stage (Create) where they were free to create any type of projects as a final assignment. They used this as an opportunity to demonstrate how to apply the knowledge they have learned from the tutorials. Although students were given planned class time to participate in the activities throughout the three sub-stages, they were welcome to go on Scratch on their own time at home. For my Montessori students, they had the choice to work on Scratch activities after they have completed their work cycle during Independent Work blocks. For my students in the Neighbourhood program, they were allowed to use their personal devices to access Scratch during Flex blocks, which were 30-minute blocks of time after lunch where they could work on any school-related assignments.

These three sub-stages —Discover, Learn, and Create — were designed to simulate how a Montessori material is typically used in the classroom to learn a new concept. With traditional materials, students are first given a presentation by the teacher on how to properly use a material before working with it independently. Similarly, the Scratch tutorials provided students with presentations on how to use different blocks to code a program. Moreover, this framework resembled the three-period lesson structure that is one of the central tenets of Montessori education (Montessori, 1912). Although the three-period lessons are predominantly used in the primary level for vocabulary acquisition, this structure aids students with association, recognition, and recall (A.S. Lillard, 2005).

Stages 2 and 3: Data Collection and Data Analysis. As illustrated in **Error! Reference source not found.**, the data collection stage occurred concurrently with as well as after the first stage. During this stage, field notes on how students interacted with Scratch and how they interacted with one another were taken and recorded. At the end of the Create sub-stage, I collected additional data through the use of small-group interviews and individual surveys to extract students' opinions on their experience with using Scratch. After these data were collected, they were coded and analyzed for any underlying themes.

Methodologies

The methodologies used in this study is qualitative and quasi-quantitative, and differ slightly between the two groups of participants in terms of how data were collected. For the first group of participants, I used Evernote to record my observation notes whenever students were participating various Scratch activities. In order to collect data on students' experience with using the program, I conducted small group interviews and recorded them using a voice recorder. Knowing my students would divulge more detail with their opinions in environments that they feel safe and familiar in, the interviews were informal in nature and resembled check-in sessions that they were accustomed to whenever they work on projects. These interviews were done in the common space in the library that was attached to the computer lab where the rest of the students worked. Groups of 3 to 4 participants were selected based on their social connection with each other so that they feel more comfortable in sharing their opinions with me. Figure 3 shows the set of questions used in each of the interview sessions for the first group of participants.

- What do you think about Scratch?
  - What did you like about it? (What made Scratch fun?)
  - What did you not like about it?
- What was the most frustrating thing about learning and using Scratch?
- How is using Scratch similar to or different from using traditional Montessori materials to learn new concepts, such as Math?
- How do you know when something is not working correctly for you in Scratch? What would you do?
- Do you spend time to take a look at or play around with other people's projects?
  - If so, does it help you with learning how to code? How so?
  - If so, what do you find out from other people's project?
- Did you ever comment on someone else's project? If so, what kind of comments do you leave?
- Has anyone left comments for your project? If so, what kind of comments did you get? How did you feel when people leave you comments?

Figure 3. Set of questions used to gather data about students' experience with using Scratch (first group).

For my second group of participants, I slightly changed how I collected my data. Again, observation notes were taken, but formal data was obtained through written records rather than interviews. At first, I intended to only use a survey (Appendix C) that was created as part of a guided Writers Workshop lesson on how to write a review as my only data source. However, I was surprised to discover that the input provided by my students in their Scratch Hackathon self-evaluation forms (Appendix D) contained information that was also valuable for the study. The surveys were completed by students in Microsoft OneNote while the self-evaluation forms were completed on paper.

To prepare for data analysis, the audio recordings were transcribed and the transcripts from individual questions were extracted. Together, with the written data from the surveys, they were coded into the following categories:

- Follows the planes of development
- Isolates different concepts
- Sequences from concrete to abstract
- Fosters collaboration
- Provides control of error
- Motivates intrinsically
- Promotes peace
- Fosters creativity and imagination
- Engages the senses
- Maintains learner concentration

- Empowered Learner
- Digital Citizen
- Knowledge Constructor
- Innovative Designers
- Computational Thinker
- Creative Communicator
- Global Collaborator

Data that were sorted into the first column of categories helped determine whether Scratch could be considered a Montessori material, as they are the principles of Montessori teaching and learning suggested by Prozesky and Cifuentes (2014). Data that were coded under the second column of categories are used to show how Scratch help students develop digital literacy, as they are ISTE's 2016 *Standards for Students* that MediaSmarts suggests to use to measure and assess digital literacy skills. Within each category, a "positive/negative/neutral/irrelevant" indicator was used to denote how student feedback on using Scratch contributed to the criterion. All of the coded data were recorded using Microsoft Excel so that they could be sorted and filtered with ease during the analysis stage. As an extra step, these data were also quantified and summarized in graph format.

Ethical Considerations

Since the participants of this study involved children, there were issues that I needed to consider carefully in order to ensure I performed my research activities ethically. This is important to preserving and respecting the dignity, well-being, and rights of my students (Graham, Powell, Taylor, Anderson & Fitzgerald, 2013). An analysis of the necessity of my research and the potential benefits and risks that may arise from my research are outlined in the following sections.

Necessity of the research. Before beginning any of the research processes, it was important to reflect on whether it is necessary to carry out my research. While there are existing publications on advocating the use of digital technology in a Montessori classroom (Boyd, 2008; Love & Sikorski, 2000; Powell, 2015) and promoting programming skills in children (Peppler & Kafai, 2005; Pierce, 2013), there has not been a connection between the two. With the development of a new curriculum by the BC Ministry of Education, there is a new focus on programming across the K-9 grade levels (BC Ministry of Education, 2016). Therefore, I used the opportunity to inquire whether Scratch possesses the properties of traditional Montessori materials and as such be considered as one for developing digital literacy skills. It could help strengthen the case for opening up opportunities to integrate other digital and media technology tools into the Montessori classroom and classifying them as new types of Montessori materials. Therefore, a study on the experiences of students learning how to code using Scratch was necessary in order to collect data for analysis.

Potential benefits. Besides helping me achieve my goal of collecting data for my research, both groups of students in my study also benefitted from their participation. For instance, by learning how to code in Scratch, students were able to practice logical thinking skills in a fun and creative way that also allowed them to claim ownership of their work, something that Pierce (2013) argued in his work. Moreover, students gained valuable skills in core principles of design that Resnick (2012) believes could transfer across various disciplines and activities. Lastly, Scratch promotes exploratory, hands-on learning and provides instant sensory feedback as users learn to code. These not only align with the essence of Montessori education but also intrinsically motivate my students to learn.

Potential issues to consider. Besides benefits, there were issues that I needed to consider and address throughout my research in order to minimize risks that could affect my participants. These issues surround consent and assent, district policies, inclusivity, privacy, and potential harm.

Consent and assent. Since the activities my students participated in were part of my explicit curriculum as well as part of my students' routines (check-ins and self-evaluations), I was not required to obtain parental consent and participant assent for activities that the students took part in. However, in order for me to gather formal data (interviews and surveys) to answer my inquiry for this paper, I needed to inform my students and their parents about the study and to obtain proper consent and assent. According to Canada's Interagency Advisory Panel on Research Ethics (2014), there is not an established age of consent for children; rather, it is the researcher's responsibility to determine whether participants "have the capacity to understand the significance of the research and the implications of the risk and benefits to themselves." Due to time constraints, it was not feasible for me to explain my study in detail and in a way that my students could understand easily. Thus, I decided to seek formal consent (Appendix E) from my students' parents to use their data for the study. However, I also obtained my students' assent after informing them that they were free to withdraw from giving me permission to use their feedback on their experience using Scratch at any given point in time during the unit.

District policies. The school district in which I work supports teachers in using cloud tools in their classrooms to enhance their teaching. However, for any digital tools such as Scratch that store student account information in servers located outside of Canada, permission letters must be sent out to parents as “student privacy and safety must supersede the ease of use and accessibility of any digital tools” (Cloud Tools, n.d.). These letters provide parents with information about how a selected tool would be used, how student-generated content would be shared, how student accounts would be generated and deleted, as well as a link to the digital tool’s Terms of Use document. A copy of the permission letter I sent to my parents for Scratch can be found in Appendix F.

Inclusivity. One of the biggest ethical issues that I faced when conducting this research project was how to ensure participation is inclusive. For my first group of students, they were given one hour each week to work on Scratch activities in the school’s computer lab. In order to maximize participation from as many students as possible, I needed to take into consideration the fact that some of them received pullout student services support throughout the day on a regular basis. Thankfully, I was able to locate a block of time from 2:00pm to 3:00pm every Wednesday afternoon when I have most of my students present. However, there were still three students who were not able to participate in any Scratch activities in-class: one was a high-level athlete who only attended school in the morning, one received EAL pullout support every Wednesday afternoon, and another student who was designated with intellectual disabilities and was on a modified schedule. For the first two students, there was a potential risk that they might feel disconnected when their peers talked about their experiences with using Scratch outside of class. To alleviate this, these two students were invited to participate in Scratch activities on their own time at home and were given the same opportunity to participate in my data collection process. However, for the third student, it was not feasible for me to ask him to do the same independently without adult support and therefore was not included in my study.

For my second group of students, this was not as big of an issue because there was only one student who received student services support in-class. Also, most of them brought their personal laptops to school to use for Scratch activities, and there were enough school-owned device to loan to those who did not. As a result, there was a lot of flexibility for when students participate in Scratch activities in order to maximize participation.

Privacy. There were several steps that I took to protect the privacy of my students during their participation in my research. First, when creating student accounts in Scratch, no identifiable information was embedded in the usernames; I use pseudonyms to identify each student and created accounts using them. I also reminded my classes about not including any personal information when creating content in Scratch, as well as not to use photos of themselves as profile pictures for their accounts. Secondly, theses pseudonyms were used when I coded my data and when I quoted my students directly as ways to protect their identities. Lastly, all of the student accounts in Scratch were deleted at the conclusion of my research, along with any created contents.

Screen Time and Feedback. Another issue to consider was the potential physical and social-emotional health-related risks that my students could be exposed to due to increased screen time by participating in my research, especially if they decided to work on their projects outside of the prescribed class time. To address this, conversations with students about self-regulation and maintaining personal well-being concerning technology use (such as taking breaks away from the screen) occurred throughout the duration of their participation with Scratch. Social-emotionally, since Scratch promotes collaboration through community, students were bound to receive comments on their work from their peers as well as other online users. Some of these comments, such as ones that deliberately aim to provoke or upset others, could cause a negative impact on my students' social-emotional health. To deal with this, mini-lessons on how to handle criticism, how to leave constructive feedback, and how to be socially responsible in an online environment were given prior to students sharing any of their work with each other and within the Scratch online community.

Findings and Interpretations

In total, 44 students participated in my study over a 12-month period from March 2016 to February 2017. As mentioned in the Methodologies section, the data I collected from them consist of field notes, observation records, interviews, and surveys. Some of these data were quantified, thus making my study both qualitative and quasi-quantitative in nature. Since my inquiry concerned whether learning how to code in Scratch helps student develop digital literacy skills and whether Scratch could be considered as a Montessori material, I have decided to use ISTE's (2016) achievement indicators for digital literacy as well as Prozesky and Cifuentes' (2014) principles of Montessori teaching and learning to interpret my data.

Scratch and ISTE's Achievement Indicators for Digital Literacy

ISTE in 2016 established its *Standards for Students,* which contain seven profiles – each with a set of achievement indicators – to assess student competency with using digital technology for learning. These indicators are presented in the form of roles that students can assume as they learn in the digital age. The seven profiles are: Empowered Learner, Digital Citizen, Knowledge Constructor, Innovative Designer, Computational Thinker, Creative Communicator, and Global Collaborator. Figure 4 provides a graphical summarization of these roles.

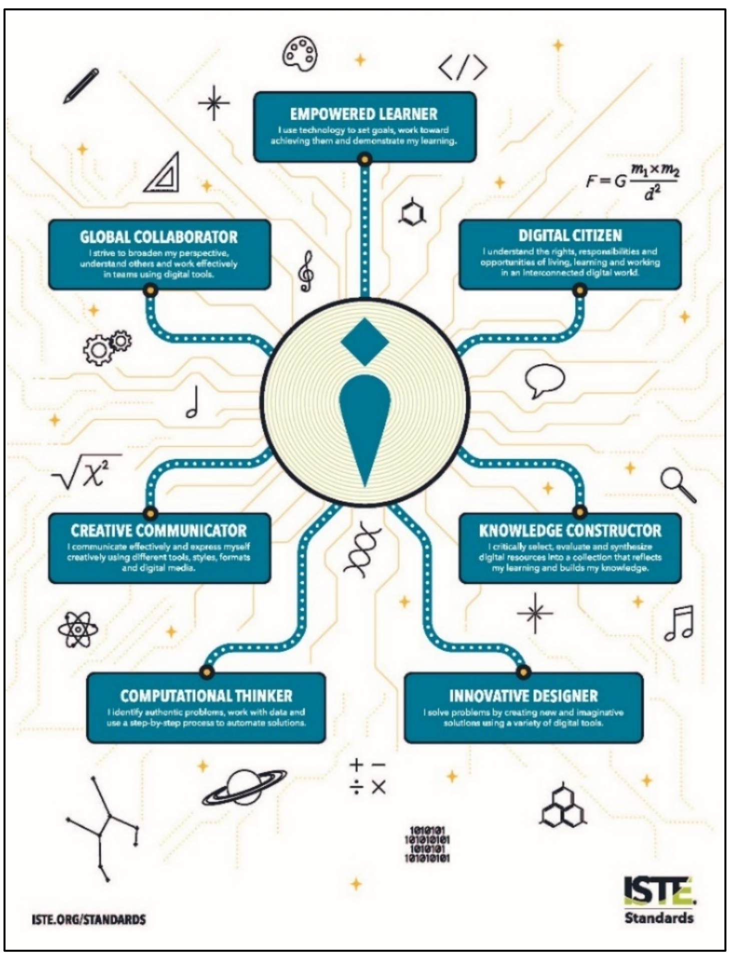


Figure 4. I am a Digital Age Learner Poster retrieved from ISTE Standards for Students, 2016.

Empowered Learner. After reading all my students' self-reflections on their Scratch Hackathon project, a number of them expressed positive thoughts on programming and how their experiences empowered them as learners. Many indicated learning, experimenting, and trying new things with coding as their highlights, while some felt gratification when they were able to share their products with their peers. Participants also enjoyed establishing ownership over their programs and developing other skills besides programming skills over the course of their project. The following are excerpts were taken from their reflections, which align with the Empowered Learner profile:

> *"Being able to create a game was so much fun and especially getting to see the end result." (2NA02)*
>
> *"It was cool how we made the game our own." (2KB05)*
>
> *"…how to problem solve more efficiently so that I could learn a lot better and faster." (2MC06)*
>
> *"Learning how to code = highlight…I could do a lot more than I thought." (2MC07)*
>
> *"Hackathon helped me learn how to balance multiple tasks and jobs." (2LF10)*
> *"…having the feeling at the end when we realized that we had actually made a game." (2NF11)*
>
> *"What I liked about our shared project was that lots of people were interested in the ideas of our project and they liked the game." (2KL17)*
>
> *"A highlight from my Hack experience was when people who played our game said they really enjoyed our game. It made me feel like all the work throughout the past 3 weeks developing the game was worth it." (2NM21)*
>
> *"I got to know [2AS24] better." (2JM22)*
>
> *"A highlight of my experience is getting to share to the other classes." (2IP23)*
>
> *"I like [our game] attracted a lot of people…I am confident that I can make things in Scratch." (2AS24)*

Digital Citizen. As a Digital Citizen, learners understand "the rights, responsibilities and opportunities of living, learning, and working in an interconnected digital world" (ISTE, 2016). Although there weren't any hard data reflecting how learning programming in Scratch directly aids in developing this competency, it is worth noting that my students received mini-lessons and participated in class discussions throughout the study. For instance, participants in my first group were allowed to create their own accounts, as class accounts for Scratch was not available as a feature at the time. We took time to talk about what is appropriate for a username for school work, as well as why it is not safe to include photos of themselves on their account profiles and in their projects. For both participant groups, we discussed etiquettes for commenting on other people's projects on the Scratch website. This included the use of appropriate language and how being anonymous does not give users license to be irresponsible for their actions. Lastly, although Scratch encourages users to remix projects – adapting and modifying another user's code to make it their own – I had a conversation with my students about the responsibility of giving credits where it is due (Figure 5). We also had lessons on licenses on copyright materials that students might put into their projects such as pictures, artwork, and music.

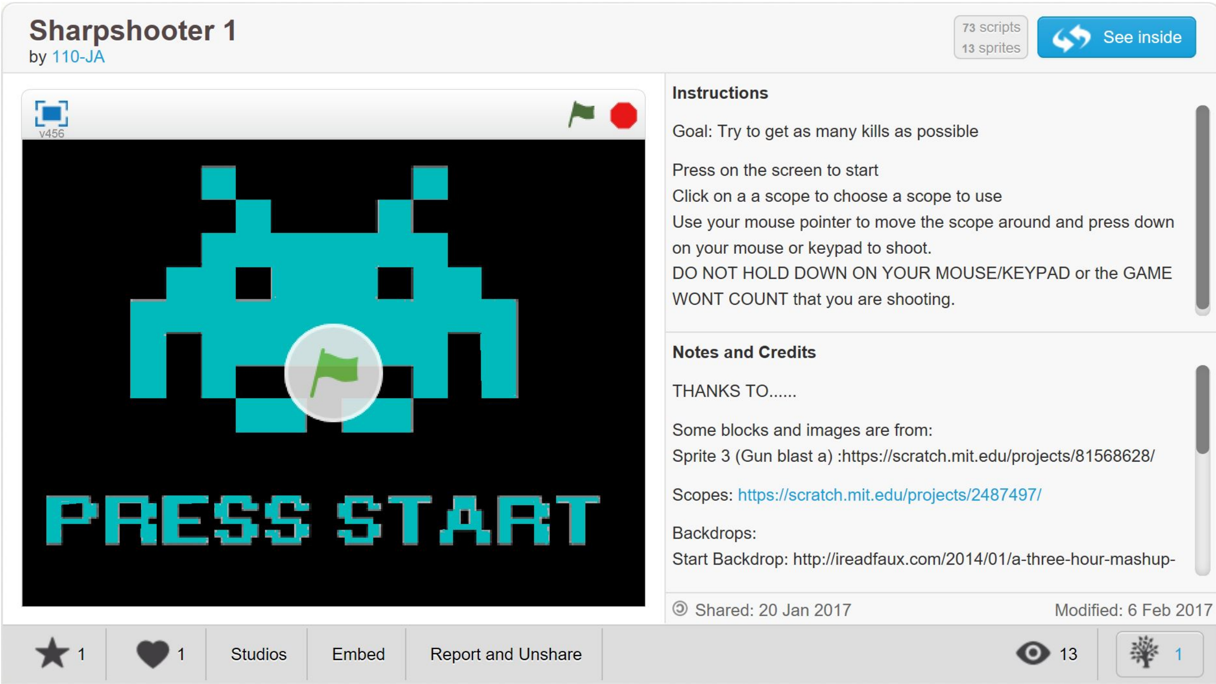


Figure 5. Screenshot of Scratch Project Description. Students gave credits for using any copyrighted materials in their project.

Knowledge Constructor. Since most of my students who participated in the study have had no prior experience with using Scratch nor with computer programming, it was necessary to spend more time during the Explore and Learn sub-phases before students use Scratch to create their own projects. Nevertheless, the amount of time available to me for conducting the study was limited, and thus so is the time the students had for fully understanding how the different Scratch blocks work. During the Create sub-phase, some students struggled to get their programs running the way they wanted and began to feel discouraged. Fortunately, since Scratch is built for users in different communities to share ideas, my students were able to browse on Scratch's website and take a look at what others have done. Moreover, users can go inside programs and explore what blocks were used and get inspirations for their own projects. As mentioned by some of the participants in my first group during informal interviews:

"I look at the games and try to look at it and see what you can do." (1TM)

> *"I like how you can see other people's projects and play [them] because it's pretty amazing kind of stuff that other people can do." (1CJ)*

> *"I look at other people's work…finding different ways to do something." (1JS)*

To me, my students demonstrated a Knowledge Constructor's ability to "select, evaluate, and synthesize digital resources into a collection that reflects [their] learning and builds [their] knowledge" (ISTE, 2016). They *selected* from a wide variety of projects to look for inspirations and *evaluate* which ones were useful for their own projects. Throughout the process, students *built* their knowledge of coding and *synthesized* what they have learned in a project that *reflected* their learning. In a way, I was glad to have such a limited amount of time that my students had to do all these out of necessity and not because I told them to do.

Innovative Designer. Given the limited amount of time my students had with Scratch, their projects were far from being complete and polished pieces of work. Nonetheless, they demonstrated how they could "solve problems by creating new and imaginative solutions using a variety of digital tools" (ISTE, 2016). For example, Scratch provides users with a cornucopia of sprites and sound clips that could be used in projects. Not only could sprites and sound clips be edited, users can design and create new ones by using the built-in editors (Figure 6). In some cases, my students chose to import other images they have found online to incorporate as sprites as they did not find any of the default images and backgrounds useful, nor did they possess the skills to draw their own. An example of this is a game that two students created called "Hamilton Duels" where they wanted to reenact the famous historic duel between Alexander Hamilton and Aaron Burr in the form of a video game. These students did an exceptional job at using a mixture of built-in graphics as well as imported graphics and sounds to create an original game for their peers to play (Figure 7).

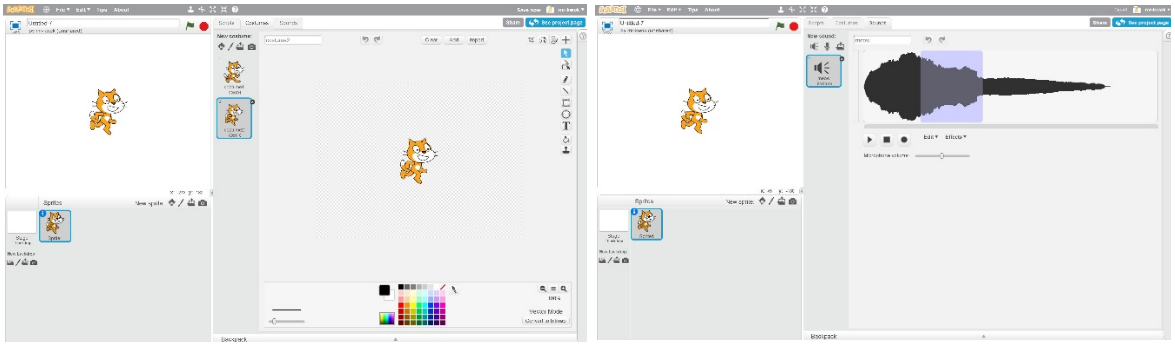


Figure 6. The Sprite Editor (left) and the Sound Editor (right) in Scratch.

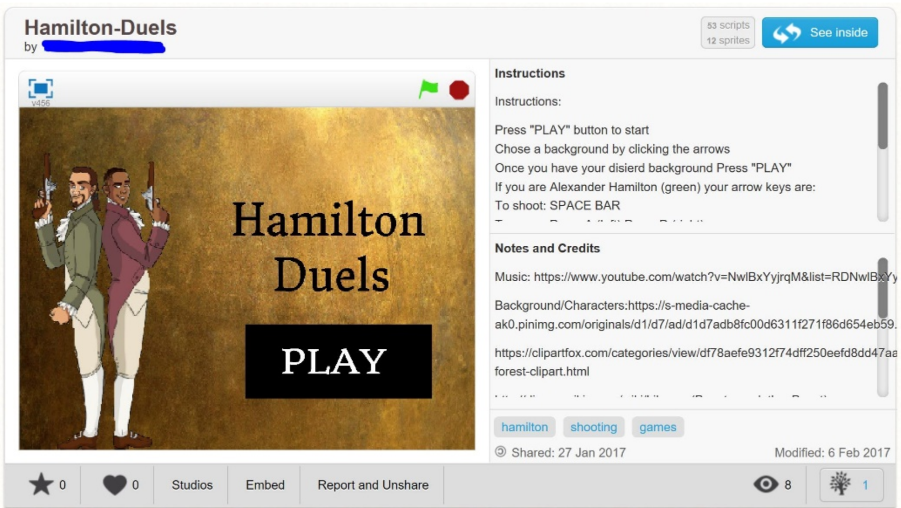


Figure 7. An interactive game designed by two students on the historical duel between Vice President Alexander Hamilton and General Aaron Burr.

Besides allowing students to be innovative, Scratch also helps facilitate the design process by providing opportunities for other users of Scratch to comment and give feedback, which authors can then take and make improvements to their projects. Although my students did not use this feature, they went through multiple rounds of sharing their work in progress with each other in hopes of gathering suggestions and ideas to further their designs (Figure 8). Our class also invited three classes from our pod who have not done the project to act as outside testers (Figure 9). As one of my students (2KJ15) said, “[the] feedback that was given was very helpful for us improve our game.”

FEEDBACK BY	[RED] What is something that doesn't work or could be improved?	[YELLOW] What is something that is confusing or could be done differently?	[GREEN] What is something that works well or you really like about the project?

Figure 8. Recording chart for gathering peer feedback on students' work in progress.

Figure 9. Students sharing their projects with their peers and gathering feedback to improve on their work in progress.

Computational Thinker. Having my students use Scratch to learn computer programming allows them to be developed into Computational Thinkers who "work with data and use a step-by-step process to automate solutions" to authentic problems (ISTE, 2016). The step-by-step automation, referred to as algorithms, is evident in how programs are created using Scratch as individual blocks denoting a single, simple action each can be stacked together to form more complex instructions for sprites to follow (Figure 10).

Figure 10. A set of instructions created using a series of single-action blocks for a sprite to automate.

However, computational thinking, according to Krauss and Prottsman (2017) requires three more fundamental skills besides algorithm: abstraction, decomposition, and pattern matching. My students practiced these skills throughout their time using Scratch. For instance, two of my students created an air hockey game as their final project. They received inspirations from a tutorial they previously completed in the form of a pong game (Figure 11). In their case, they were able to extract the concept of the pong game and adapt it into their air hockey game, which is an example of removing specific details to create a solution that works on a generic problem (abstraction).

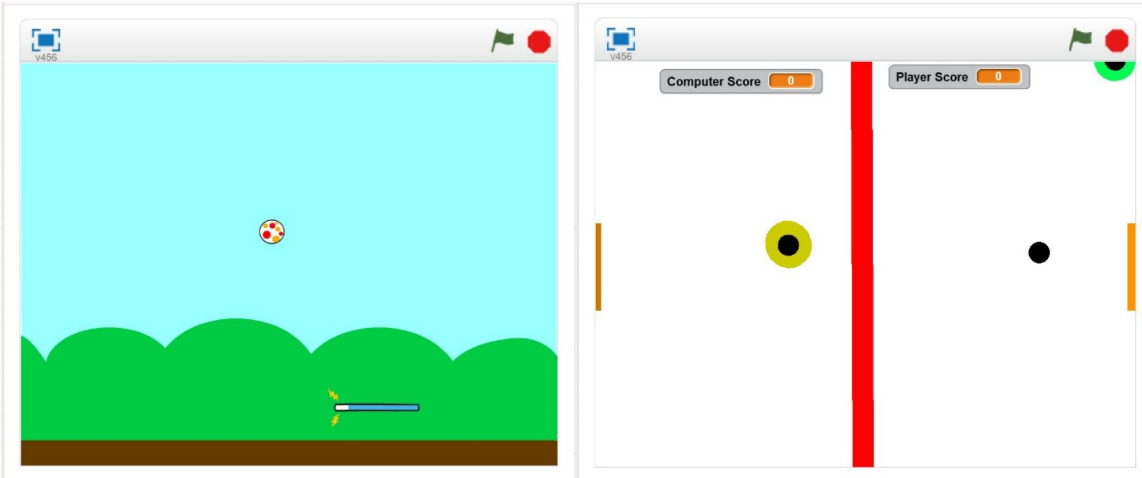


Figure 11. A pong game tutorial and an air hockey game created by using the mechanics of the pong game.

In another example, many bugs occurred in students' code when expected outcomes did not occur given an input. In these situations, the programs needed to be debugged. To do so, it was necessary for students to read through their program and break it (decomposition) into logical pieces so that they could locate the bugs. There are generally two types of bugs in Scratch, either when an incorrect block is used or when a block is used in an incorrect location. To determine the type of bug that was occurring in their code, students might recall situations in which similar errors happened in the past or any of the debugging tutorials that they have completed prior to starting their projects (pattern matching).

At the end of our Hackathon unit, my students reflected on how Scratch helped them become Computational Thinkers:

"I think that I have definitely improved on my problem solving skills…" (2SJ14)

"I learned how to debug programs and find/solve problems in a game." (2KJ15)

"I improved on problem solving." (2KL17)

"I got a lot of debugging skills throughout the project." (2BM20)

“We used our critical thinking skills to figure out which scripts to use in order to make a working scoreboard.” (2NM21)

“I think I have acquired troubleshooting skills.” (2DS25)

“I think that scratch (sic) is a good tool to use because it will help you with your creative thinking skills and your problem solving skills.” (Anonymous)

“scratch (sic) is good because it forces users to critically think about where to place scripts in order to create a game.” (Anonymous)

Creative Communicator. Due to limited time, the scope of students’ Scratch projects was also restricted in some ways as almost all my students decided to use Scratch to create games. However, there is way more to Scratch that allows users to “communicate effectively and express [oneself] creatively” (ISTE, 2016). A look at the Explore section on the Scratch website (Figure 12) gives an overview of a plethora of projects created by Scratch users worldwide, which included but are not limited to artwork, eCards, games, music videos, programs, stories, and tutorials.

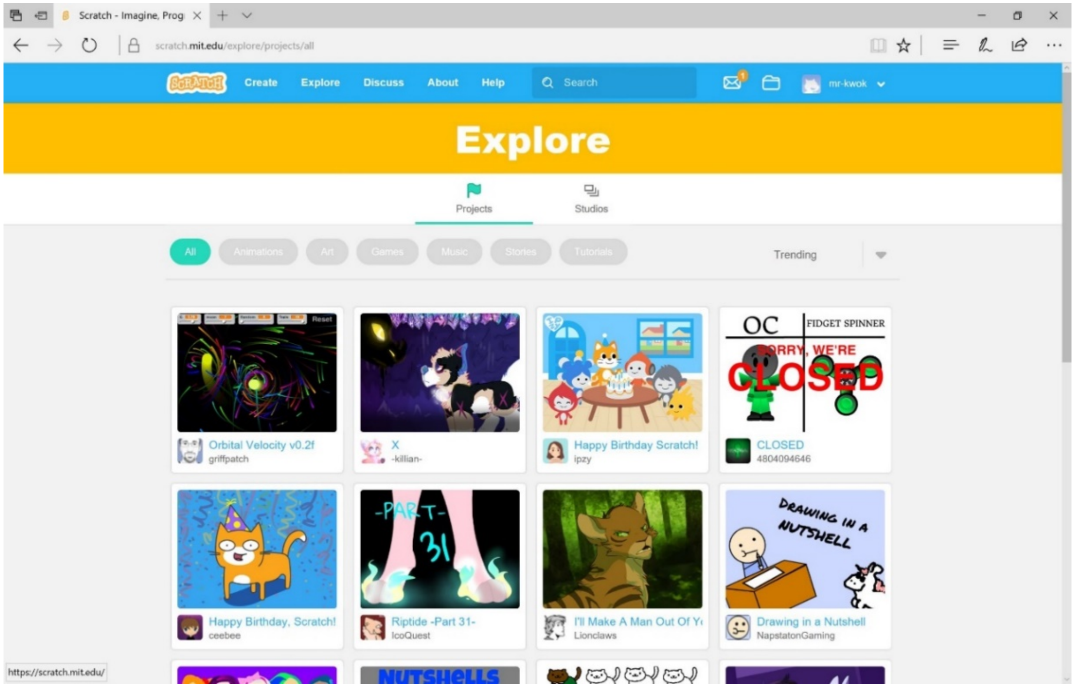


Figure 12. The Explore page of Scratch's website shows a variety of types of projects created by users worldwide.

Thus, Scratch allows students to use a variety of ways to express and communicate their ideas with others. As some of my students noted:

> *"Yes, there is a wide range of what you can do with scratch (sic)." (Anonymous)*
>
> *"You start with a blank canvas and once you learn how to use the program you can create whatever you want, and it differs from creating something really complex to something really easy. It's really up to the user." (Anonymous)*
>
> *"...you can literally make whatever you want." (Anonymous)*
>
> *"Scratch allows people to be extremely creative, as it does not limit what they can do or create to just games. Scratch can be used for many purposes. If you're an artist why not make some art and publish it? If you're an animator, you can make an animation. Scratch has so many possibilities, as with the way it is designed to be an open community program that does not limit its coders to just one thing." (Anonymous)*

Global Collaborator. Not only can users communicate and express their ideas freely using Scratch, they can also collaborate with one another on projects. Since Scratch is an online community, users do not need to be in the same physical vicinity to collaborate. For example, students not only are able to work with each other in the classroom, they can also work with users around the world to create projects by participating in Scratch's online forum as well as providing and receiving feedback in the comment sections of individual projects. Moreover, Scratch provides users the ability to remix existing projects, therefore allowing collaborators to work on the same project without the need to be on the same account or sitting next to one another. Figure 13 shows a Remix Tree of a project where two of my students worked together, sometimes separately, to debug and improve their game. A remix tree records when a user takes another user's work and makes modifications and additions to it. Although my students did not get the chance to work with Scratch users outside of the classroom, they were still able to demonstrate how Scratch enabled them to become Global Collaborators by allowing them to "broaden [their] perspectives, understand others and work effectively in teams" (ISTE, 2016) (Figure 14). At the end of the projects, some of them had the following insights:

> *"I helped many people with their code." (2KA03)*
>
> *"I have learned the pros and cons while working with others on a project." (2KJ15)*
>
> *"With deadline approaching there were many bugs so I went to my group and asked how I could solve the bugs." (2MC06)*
>
> *"I like that we got to share ideas and collaborate with others." (2RG12)*
>
> *"I got to interact with others and work together as a team to solve [problems] and create programs." (2JM22)*
>
> *"If you spend time trying to figure it out then you can head over to the Scratch Forums and find help with other members of scratch (sic) community." (Anonymous)*

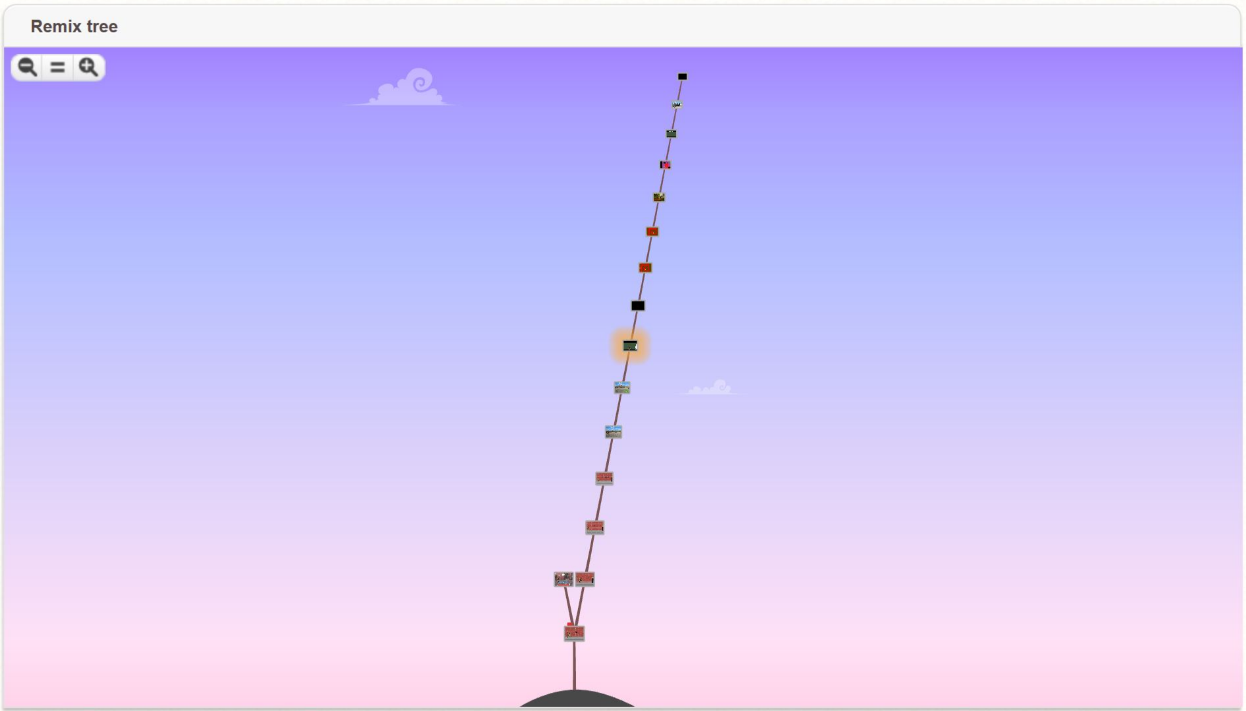


Figure 13. A remix tree showing iterations of development of a Scratch project.

Figure 14. Students collaborating on a Scratch project.

Further Readings. Although my case studies demonstrated how Scratch helped my students develop digital literacy skills by categorizing my observation and interview and survey data using ISTE's 2016 Standards for Students, there are many scholars in academia who also championed the benefits of using Scratch in an education setting. Peppler and Kafai (2005) performed a study and argued that Scratch could be used to teach creative coding which they felt is "essential to expression in a digital medium" (p. 1). Meanwhile, Monroy-Hernández and Resnick (2008) discussed how Scratch "offers an alternate model for how children might use the Web as a platform for learning" (p. 50). Others, such as Kalelioglu and Gülbahar (2014) and Su, Huang, Yang, Ding, and Hsieh (2015) suggested how Scratch could help students develop problem-solving skills.

Scratch and Montessori Principles of Teaching and Learning

In order to determine whether Scratch could be considered a Montessori material, it is necessary to see how it fits in with the Montessori philosophy. Prozesky and Cifuentes (2014) argued that Maria Montessori was a user of technology as she created it to help students learn, and that "though digital technologies were not available when Montessori was developing her theories and methods, the Montessori philosophy can still inform best practices when it comes to the use of technology in the modern classroom" (p. 30). The authors also provided ten principles of teaching and learning for technology integration in the Montessori classroom which I used to argue how Scratch can be considered a Montessori material, supported by student interviews and field observations.

Engaging the senses. Maria Montessori (1912) believes that for students to learn effectively, it is necessary that their senses be engaged. The Scratch programming environment is able to support this in several ways. First, individual blocks of instructions for sprites are organized into ten different categories, with each category denoted by a colour. Traditional

Montessori materials are designed with colours in mind to engage students. For example, the number beads and the board that are part of the Checkboard material used for multiplying are designed using different colours. The beads denoting numbers from 1 to 10 are arranged in unique colours while the checkboard uses the colours blue, green, and red to signify specific place values (Figure 15).

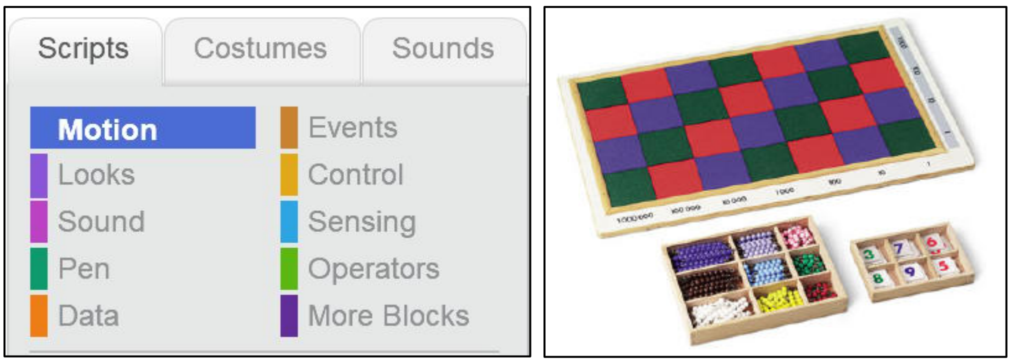


Figure 15. The use of colours in Scratch (left) is similar to the use of colours in traditional Montessori materials, such as the Checkerboard material (right), to engage the visual sense in students.

Besides visual cues, Scratch also engages one's hearing sense through the provision of sound effects that users could use in their projects. This is done using Sound blocks (Figure 16). Scratch offers users a wide range of sound clips that they could sample before inserting into their script by simply double-clicking on the appropriate blocks to activate the sounds. I noticed that upon discovery of the Sound blocks, both groups of my students showed a great deal of enthusiasm in exploring and playing with the audio clips, with some of them even attempted to record their own using the program's built-in Sound Editor. On the other hand, one feature that I wish Scratch could integrate into its programming environment is audio feedback when users insert blocks into their scripts. Other programming environments, such as the JavaScript Blocks Editor for coding micro:bits, offers sound effects to let users know that they have successfully “snapped” in a block of code. This would be a useful feature to engage students in a different way as well as provide a clear control of error.

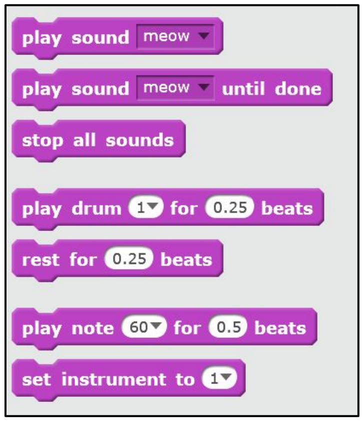


Figure 16. An example of Sound blocks that users can insert into their script.

Another feature that Scratch contains is the ability to sense user input in the form of video motion and loudness through a computer’s webcam and microphone respectively. When these Sensing blocks are used, programs can be created to allow for user interaction using body movements and voice input, which are ways to engage other human senses (Figure 17).

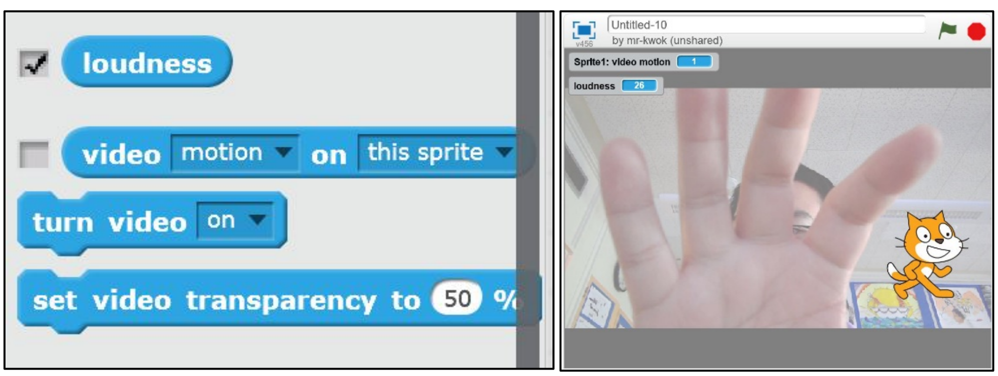


Figure 17. Video and audio Sensing blocks in Scratch (left) and an example of real-time sensing when they were used (right).

Despite how Scratch is able to engage the senses of learners visually, aurally, and kinesthetically, there was mixed feedback from my students when asked about whether the user interface of Scratch enticed them to use the software to learn about programming. Figure 18 shows the summary of my students’ responses. Of the 26 students who participated in the survey, only 7 thought that the programming environment in Scratch engaged them to learn coding, while 10 of them reported negative engagement. The following is a sample of their written responses:

> *"The interface I believe, does engage the users to learn to code, as scratch provides an easy to understand layout, with the blocks of code separated into different categories based on what they do. The way it is set up encourages people to try out different blocks of code to see what it does, and to experiment from there. The blocks of code are also color coded by function, making it easy to tell which category they are under."*

> *"For me, Scratch is appealing because most people (including myself) are used to coding platforms such as JavaScript. JavaScript uses letters and numbers for coding, and to beginners it just looks really complicated. Scratch, however, uses blocks that you drag to code."*

> *"Simple, colorful, attention grabbing…Better for catching younger audience's attention."*

> *"The interface does not draw my attention and make me want to code because it looks a bit plain to me, with just the scratch cat there in the middle it doesn't make me want to code."*

> *"By the look of the interface its (sic) not that appealing to me and it doesn't intrigue me to want to learn more on how to code…When you look at the interface it kind of looks boring and it doesn't look like you can create a working game…the whole layout of scratch didn't draw my attention either."*

> *"Get a script, drag it to a box to make it do something, repeat. Doing this for hours will make you look like a zombie and not a (sic) engaged learner."*

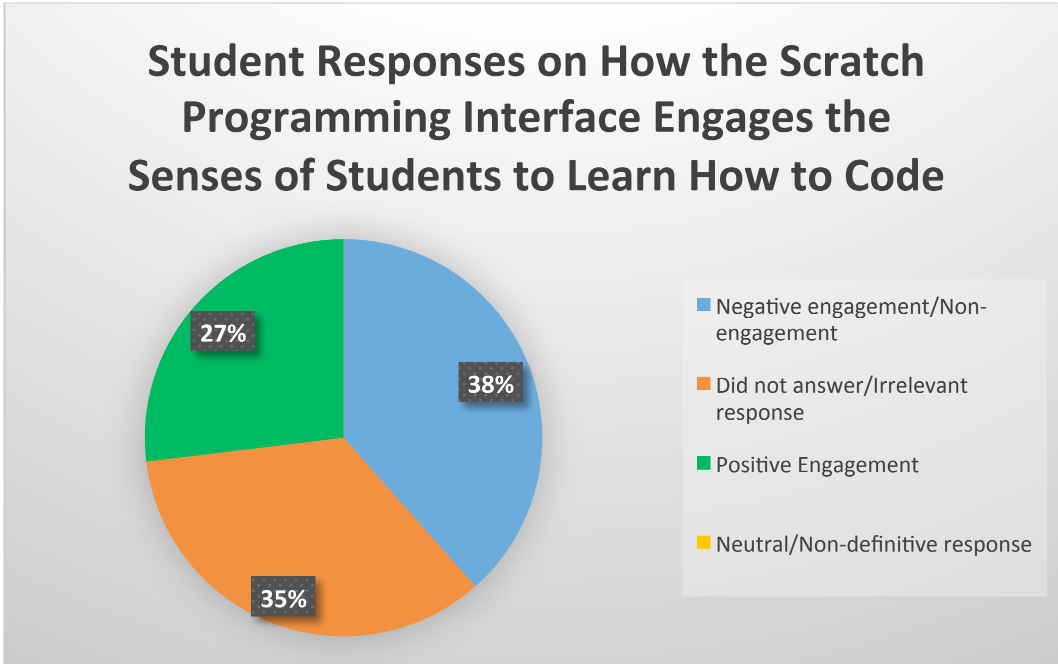


Figure 18. Summary of responses on how the Scratch programming interface engages the senses of students to learn how to code.

Control of error. One of the tenets of the learner-centred pedagogy that Maria Montessori based her philosophy on is that teachers are guides and facilitators. Thus, the materials that Montessori students use need to have self-correcting mechanisms that provide immediate feedback and encourage students to work independently. According to Prozesky and Cifuentes, "well-designed and implemented technologies are a good match for a learner-driven environment such as a Montessori classroom" (2014, p. 32). There is evidence that Scratch is able to achieve this: Although there are many blocks that users can use in Scratch, not all blocks can be used with one another. Like jigsaw puzzle pieces, different types of blocks have different shapes. These shapes will fit together only if a jutting bottom edge of one fits into the "trough" of another on the top edge. This gives users a simple way to know which blocks do not make sense — for example, an action block and a data or condition block which has no juts nor troughs — when put together, as the shapes would not fit (Figure 19). Furthermore, when a block can be used with another block, users are provided with immediate visual feedback. Figure 20 shows an *if-then* statement block that has a hexagonal slot. This tells users that only a hexagonal block, which in Scratch denotes Boolean (true or false) conditions, can be used in the statement. When a user drags a conditional block near the slot, the slot will be highlighted. Conversely, if a user drags an oval-shaped Data block near the slot that requires a conditional statement, the slot will not be highlighted. This is a hint to let the user know that it is an incorrect block to use with the if-then statement.

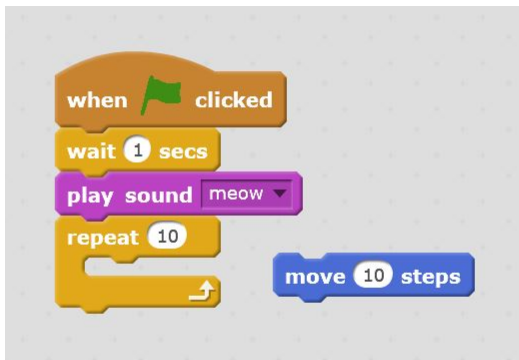


Figure 19. An example of how blocks in Scratch are designed like jigsaw puzzles as a means to control user errors.

Figure 20. Visual feedback for when a compatible, Conditional block is snapped onto an if-then block (left) and an incompatible, Data block is attempting to snap onto the same block. The shape of the slots provides an easily recognizable control of error for users.

Traditionally, when users need to test out their programs in programming languages such as C, C++, or Java, they need to first compile their programs where the computer detects any syntax errors by the programmer. Only when the program is error-free will it run. This arduous process of compiling and running the program for users to test out their work is not necessary in Scratch. Since the preview area is built on one side of the screen and the workspace for scripts on the other side, and that it is impossible for users to produce syntax errors, users can see their work in progress running as they code. This is done by double-clicking on any script in the workspace for a particular sprite, or when the green flag icon is clicked to start the entire program. Therefore, students who are learning how to program receive feedback almost instantaneously. Furthermore, students can tweak their scripts on-the-fly, which means blocks can be swapped in and out and input values changed and the program will instantaneously reflect the changes. However, one thing that I observed was that when there was a long and complex script that students needed to debug, they had a hard time pinpointing where their bugs were, as Scratch does not indicate the particular block that is running at a given time.

Overall, the visual cues and immediate feedback provided by Scratch form an effective error-controlling mechanism for students learning to code. Figure 21 shows a summary of responses from my students on their views on how Scratch help them find and correct mistakes made in their programs. Again, the result is mixed. 7 of 26 students found that they had a positive experience with the error-controlling mechanisms. The same number of students reported a negative experience, and 11 of them were either neutral or their responses were non-definitive. The following is an excerpt of their written responses:

> *"Usually when code does work it has a glowing yellow light around the script."*
>
> *"It is extremely easy to know if something isn't working because It will stand out like crazy."*
>
> *"One of the things that annoyed me with scratch was that every time there was a bug we had to take the whole thing apart and find it because it is not obvious."*

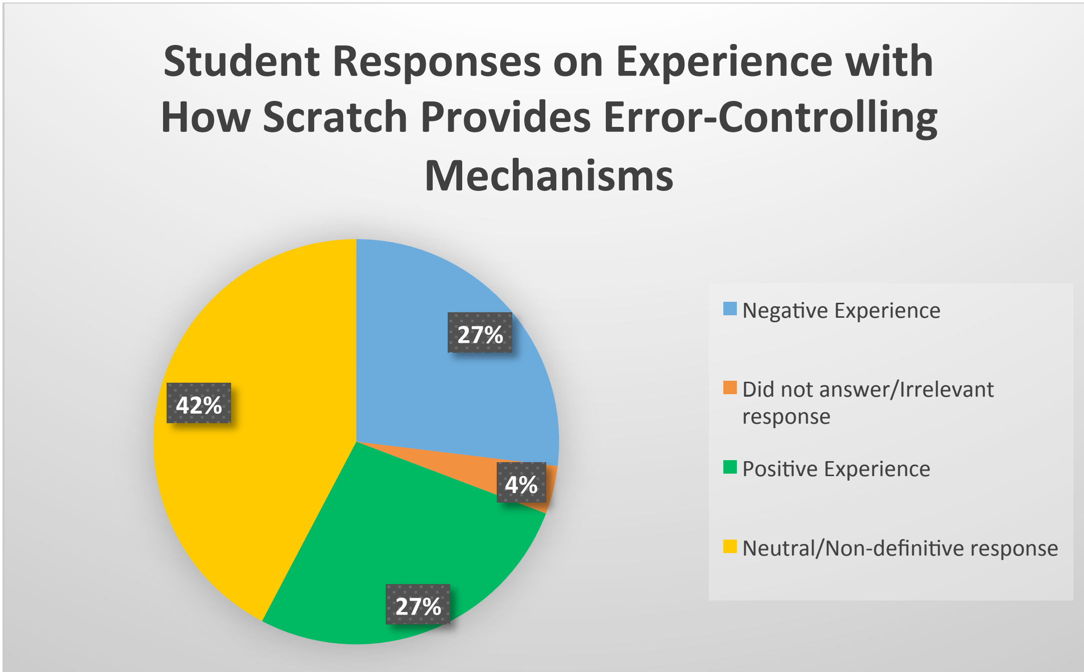


Figure 21. Summary of student responses on their experience with how Scratch provides error-controlling mechanisms.

Collaboration. Prozesky and Cifuentes claimed that technologies in a Montessori classroom should allow students to collaborate on projects and solve problems, and the concept of classrooms could be "expanded to include 'classmates' from other schools or cultures" (2014, p. 35). As explained in the previous section, Scratch not only allows for but encourage users to collaborate. Some of my students echoed this view, while some thought otherwise (Figure 22). Overall, there were 10 students agreed that Scratch encourages collaboration, but 6 students disagreed and 8 students were neutral. I suspect the reason for this is that with limited time, my students did not get a chance to use Scratch's online commenting feature to fully collaborate with one another. Also, some of them chose not to remix each other's projects that allows them to build upon one another's ideas. Instead, they relied on programming side-by-side using the same account while on multiple computers and it created many problems for them:

"You and your partner cannot work on the project at the same time since scratch doesn't save automatically and that will ruin and change your project."

"In my opinion scratch doesn't really allow you to work on projects together because it does not update onto the servers immediately. When you save your work it overlaps any work that your group has done so it makes it very hard to work together."

However, there are other students who appreciated how Scratch encouraged collaboration:

"Scratch is a great place to be able to connect with each other and view their projects. It allows you to work on the same project, however maybe not at the same time. If to (sic) are working at the same time, it might glitch out."

"I got to interact with others and work together as a team to solve (problems) and create programs."

"Together, you can design, plan, and fix problems together within your project."

"Scratch is built to be able to create a project, share it, and then people can remix it. For my last project that I made with a friend we would remix each other last version of the project."

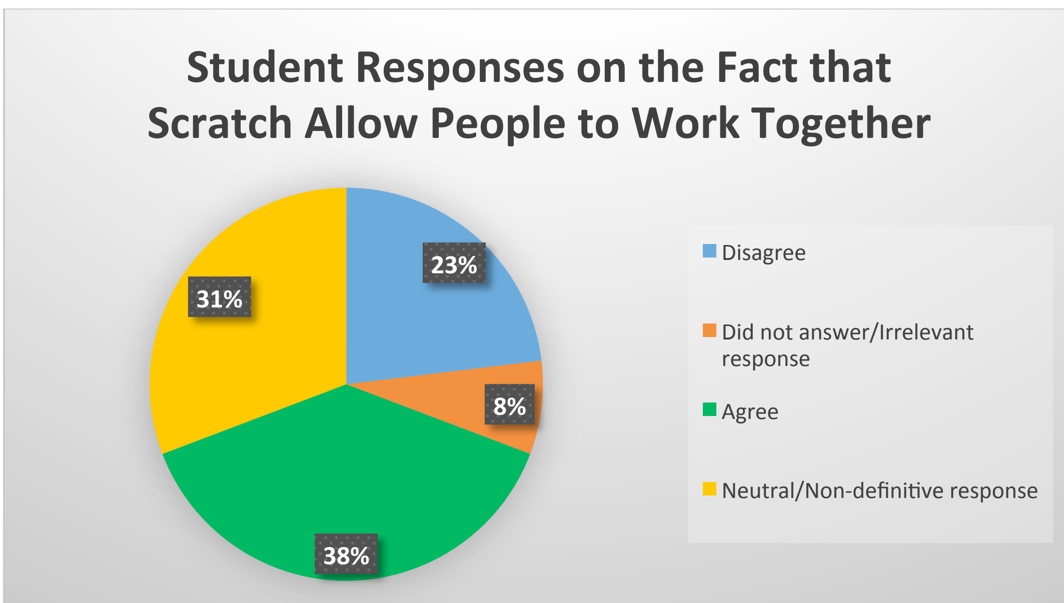


Figure 22. Summary of student responses on whether they think Scratch encourages collaboration.

Isolation of concept. Prozesky and Cifuentes (2014) argued that learning materials used in a Montessori classroom "should be designed with an isolated single concept, allowing the child to concentrate on that single factor alone" (p. 32). In Scratch, not only are blocks organized in different colours to engage the senses of learners, each of the colours denotes a different type of blocks that focuses on a different concept (Figure 23):

- The blue Motion blocks introduce students how to move and rotate a sprite, and provides opportunities for them to learn about Cartesian coordinates;
- The purple Looks blocks let students learn about graphical effects that could be applied to a sprite, such as costumes, colours, and speech bubbles;
- The violet Sound blocks feature commands that allow students to explore different sounds as well as beats and tempo;
- The green Pen blocks provide students with drawing abilities; the orange Data blocks allow for variables to be created and used;
- The brown Events blocks let students experiment with timing of running multiple scripts; the gold Control blocks teach students about conditional statements and loops;
- The cyan Sensing blocks give students opportunities to collect and manipulate any user input;
- The lime-green Operators blocks expose students to the programming language's available mathematical operations and Boolean logic statements; and finally,
- The dark purple More Blocks category lets students create customized blocks and even allows Scratch to connect to hardware devices like the Lego WeDo and Picoboard.

All in all, although programs are created using a mixture of these blocks, students who are new to computer programming can start by learning about the Motion, Looks, Sound, and Pen blocks in isolation before combining with other types of blocks to develop more complex algorithms. The isolation of concepts allows students to learn the Scratch programming language starting from concrete concepts before progressing to more abstract ones.

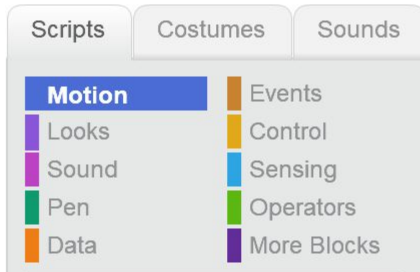


Figure 23. The different types of blocks in Scratch are organized into functions that students could learn about and practice using individually.

Sequencing concrete to abstract. All of Maria Montessori's materials are developed to allow children to grasp concepts that are "real" before moving on to ideas that are representational. For example, a young child might explore the parts of real plant before being presented with nomenclature cards and wooden puzzles, which are abstract representations, to further his/her learning with the anatomy of a plant. Programming is a very abstract skill to learn for a student from the very beginning. Fortunately, Scratch offers a scope and sequence in teaching students programming skills that is similar to how Montessori materials sequence from concrete to abstract through its tutorials. The first few introductory lessons offer ideas on how to use scripts to physically manipulate a script, such as moving, rotating, and changing size. After students have a firm grasp on these basic commands, they move on to more complex tutorials where they learn how to use *repeat* and *if-then* blocks that highlight concepts such as loops and conditional statements, respectively. Finally, the last few tutorials move students through creating variables and functions, which are some of the most advanced and abstract concepts in programming. Figure 24 shows a summary of students' thoughts on the learning curve of using

Scratch. 15 out of 26 students participated in the survey found the learning curve to be gradual or stated that it was easy to learn, while only two of them indicated the learning curve was steep. Five provided neutral or non-definitive responses.

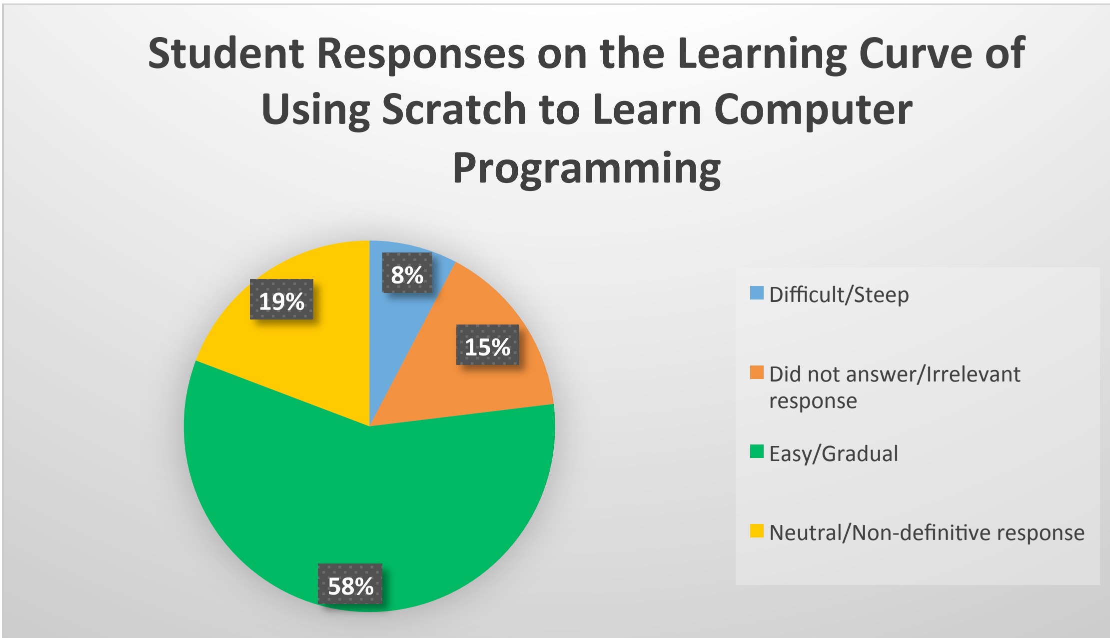


Figure 24. Summary of student responses on the learning curve of using Scratch.

Rewards and competition. Maria Montessori is a firm believer that extrinsic rewards are highly demotivating for students and thus negatively affect learning. Prozesky and Cifuentes (2014) suggested that "a well-designed learning system that allows the learner to see the big picture would be highly motivating, with progress through a sequence as its own reward" (p. 35). Scratch is able to provide this feature. Since Scratch is built to support communities of learners and various degrees of collaboration, it does not feature any sort of "merit badges" or scoring systems that encourage competitions between users. Rather, the ability to let users share their accomplishments with each other, receive feedback, and gain inspiration for creative ideas are what motivate students to learn how to program with Scratch. The following excerpts from my students demonstrated the intrinsic rewards Scratch provided for them:

> *"I enjoyed seeing what I was able to do with different patterns of code and what could help improve (my) game…I think I've gotten better at communicating, my group always discussed and addressed the issues." (2AA01)*
>
> *"Being able to create a game was so much fun and especially getting to see the end result." (2NA02)*
>
> *"I made the background from Scratch…took a long time…it was worth it and it turned out great." (2KC08)*
>
> *"Having the feeling at the end when we realized that we had actually made a game." (2NF11)*
>
> *"I was glad that people were interacting with [our game] and really enjoying playing our game." (2SJ14)*
>
> *"When we got to share our games with the rest of the team I enjoyed getting many different opinions from others." (2KJ15)*
>
> *"A highlight from my Hack experience was when people who played our game said they really enjoyed our game. It made me feel like all the work throughout the past 3 weeks developing the game was worth it." (2NM21)*

Peace. One of the foundational principles of the Montessori philosophy is peace (Montessori, 1912). Prozesky and Cifuentes (2014) discussed that when integrating technology into the Montessori classroom, "connecting learners with other learners could be a powerful tool for creating harmony in the world" (p. 35). Since students had limited time with Scratch during my inquiry, they did not have a chance to connect with other online Scratch users outside of the classroom. However, by completing their Hackathon projects in groups, some of my students felt that they have made improvements in interpersonal skills, which are lasting, transferrable, and will help them develop into peaceful citizens of the world:

> *"Some frustration was we had so many ideas that we sort of had a fight…we had a big talk." (2PK16)*
>
> *"I have learned how to get along with others and not be too bossy but still encourage them to get work done." (2AL19)*

"I developed my compromisation (sic) skills." (2NM21)

"I've improved in communicating." (2NA02)

"I have learned the pros and cons while working with others on a project." (2KJ15)

Planes of development. After having worked with and observed children, Maria Montessori (1949) established her theory of learning and development by grouping children's growth periods into "planes" of six years. She believed that students learn differently within each plane and care should be put into the development of curriculum based on the child's age. For example, while children in the first plane (ages 0-6) learn best through movement and physical manipulation of objects in their environment, children in the second plane (ages 6-12) learn best through imagination. Although my inquiry only involved middle school-aged children (the second plane) and that Scratch is primarily designed to be used by children in this plane, it is worth noting that the team from the Massachusetts Institute of Technology (MIT) who produced Scratch recently helped develop a version called *ScratchJr*, which is an application on mobile devices that allows younger children to "program their own interactive stories and games…learn to solve problems, design projects, and express themselves creatively on the computer" (ScratchJr - Home, 2017). Figure 25 shows a screenshot of *ScratchJr*, which features a simpler interface than Scratch. Since the application is built for mobile devices with touchscreens, blocks and sprites are manipulated directly with fingers. This adds a kinesthetic element to learning computer programming that is compatible to effective learning for children in the first plane of development.

Figure 25. Screenshot of ScratchJr, a simpler version of Scratch made for younger children to learn about programming.

Creativity and imagination. As mentioned previously, Scratch is a tool that allows students to be creative. However, while Maria Montessori values creativity and imagination in children, she believes true creativity "comes from generating new ideas, not simply reflecting the ideas of others" (Prozesky & Cifuentes, 2014, p. 34). The remix feature of Scratch, though able to promote collaboration, can limit creativity to a certain degree. During the Hackathon activity where my students had the chance to create anything they want, only 4 out of 13 group projects contain original ideas in terms of game design. The remaining 9 projects were either attempts to remake popular video games such as *Mario Kart,* adaptations to tutorials that have been completed, or even remixes of existing projects by other Scratch users online. I suspect the cause for this is the limited amount of time I had for my inquiry, which contributed to how long my students had for learning how to program in Scratch before having to complete the Hackathon project.

Nevertheless, as Figure 26 shows the summary of my students' response to whether they felt Scratch promotes creativity, an overwhelming 18 out of 26 students agreed and 5 provided a neutral response:

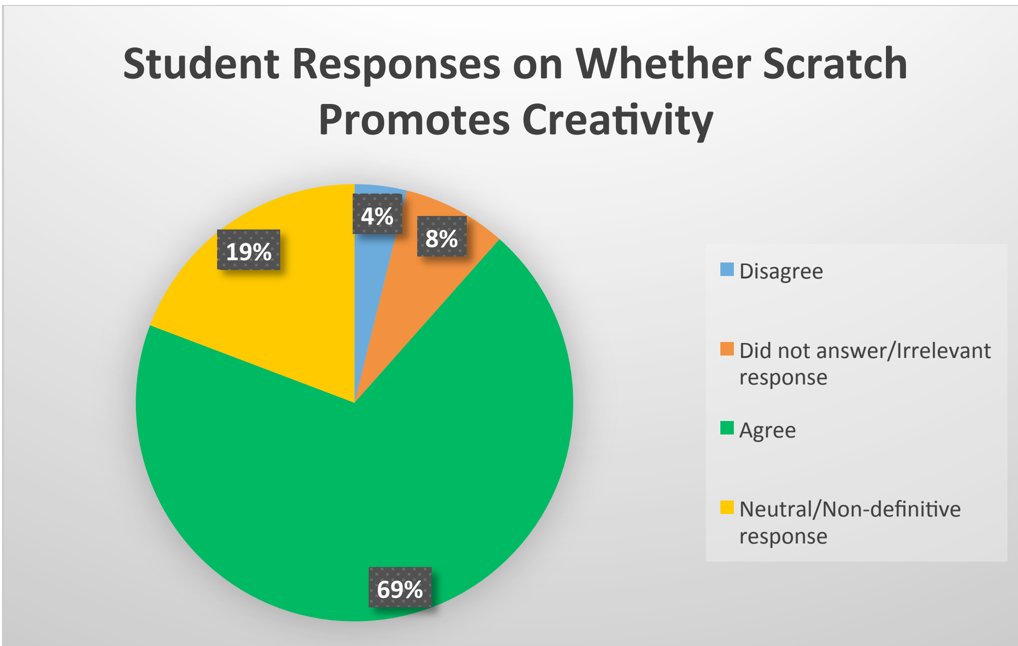


Figure 26. Summary of student responses on whether they think Scratch promotes creativity.

Concentration. One of the key learning outcomes for Montessori students is the ability to develop independence, and Maria Montessori believed that being able to concentrate on a passionate interest and focus on the associated tasks are what makes learning effective in children. For some of my students, learning how to program using Scratch was their interest, and whenever it was time during the school day to work with Scratch, they were on task and engaged in talking with one another and helping each other. For my second group of participants who were not Montessori students, some of them chose to use Flex blocks — a time after lunch when they could choose what to work on — to continue working on their Scratch projects.

Unfortunately, this type of concentration was not maintained for students in either groups. Rather than working on their projects, many proceeded to look at other Scratch users' projects and ended up spending prolonged periods of time interacting with the projects, which are mostly games. I purposely did not intervene in many occasions when I noticed this happening, since I did not want to influence the inquiry. As a result of students not using their time purposefully, they ran out of time when deadlines loomed. For my first group, many students did not finish the assigned tutorials, while students in my second group rushed their Hackathon projects, with some not being able to complete a fully-functional product. As one students said, "I was too distracted at playing [the game] to notice how it's made" (1CR).

Conclusion

As we step past the one-sixth mark of the 21^{st} century, educators, parents, and community stakeholders are viewing education from a different lens than the one used in the previous century, one that focused on fulfilling specific content objectives. The new lens shifts the focus towards a competency-based education in which students are exposed to opportunities to acquire 21^{st} century skills. In Canada, these skills are categorized around the 7 C's: character, collaboration, communication, computer and digital technologies, creativity, critical thinking, and cultural and ethical citizenship (C21 Canada, 2012). As a Montessori educator, I have shown that Dr. Maria Montessori was well ahead of the times as her philosophy of education already embraced the nurturing of most of these 21^{st} century skills in her classroom during the 20^{th} century. One particular area where there is a lack of literature is the discussion around Computer and Digital Technologies competency, also known as digital literacy. In my opinion, one of the main reasons is that using computer and digital technologies in the classroom is still a relatively novel idea compared to the history of Montessori education, and when the discussions began to surface, there were polarizing views on the thought of treating computers and other digital devices as part of the prepared environment. On one side, some Montessorians argued that traditions are meant to stand the test of time and the introduction of new technologies could disrupt the tradition and inject unnecessary distractions in student learning. On the other hand, a number of educators and researchers argued that Dr. Montessori would embrace the arrival of new technologies. Nevertheless, I believe that there must be a way we can help Montessori students develop digital literacy and become a complete 21st century learner, and it must involve the introduction of a tool as a new kind of Montessori material.

Using an established and well-adopted set of standards from ISTE (2016), I showed that teaching programming skills to Montessori students would be an effective way to develop digital literacy. Coincidentally, with the provincial curriculum experiencing a revision in British Columbia, coding has become one of the new emphases in K-12 education (Silcoff, 2016). What is missing, then, is a set of tools that could teach programming skills and also exhibit the characteristics of a Montessori material. I believe that Scratch could be an example of such a tool. Although there are existing works in academia surrounding the various benefits of using Scratch to teach programming, there is still a gap concerning how these benefits match up with the standards developed by ISTE. In order to create a bridge, a further review on Scratch literature was needed, as well as a case study involving the use of Scratch in a classroom to provide the necessary triangulation. I used the ten principles of integrating technology in the Montessori classroom outlined in Prozesky and Cifuentes' (2014) discourse as a framework to assess whether Scratch could be viewed as a Montessori material by gathering qualitative data in the same case study.

After performing these tasks, I believe Scratch could be used in a Montessori learning environment to develop digital literacy. In my case study, students gained an increasing ability to critically understand digital media content and to expand their knowledge and capacity to create with digital technology after using Scratch. Through observation and student input on their experiences, it was shown that using Scratch to learn computer programming allowed students opportunities to develop skills associated with at least one achievement indicator from each of ISTE's 2016 *Standards for Students* profile; it encouraged them to be Empowered Learners, Digital Citizens, Knowledge Constructors, Innovative Designers, Computational Thinkers, Creative Communicators, and Global Collaborators. Furthermore, although cannot be completely compared to traditional Montessori materials, Scratch demonstrated the underlying principles of Montessori teaching and learning to an extent. It was able to engage the senses of learners, provide control of error, encourage collaboration, teach concepts in isolation, sequence concepts from concrete to abstract, motivate learners intrinsically, promote peace, follow Montessori's planes of development, inspire creativity and imagination, and enable learners to concentrate on learning. Although I was able to confirm my hypothesis supported by information from my findings, there were a number of limitations to my study that may have impeded me to get more complete results and make a stronger support for my conclusion. A list of these limitations is outlined below, followed by several recommendations for any future studies.

Limitations

As I reflect on my experience with collecting data for my inquiry, there were a number of limitations that affected my project. Some of these limitations were caused by environmental factors, while some were results of my decisions to experiment with different ways to gather data. One of my biggest obstacles was the amount of time I had to gather data from my first group of participants. Although the Scratch activities my students participated in spanned three months, in actuality they were only able to spend one hour a week for twelve weeks on these activities. This is due to several reasons: First, the two schools that I carried out my study in were different in terms of the demography of my students. Socioeconomically, my second group of students had a higher status than my first group, which made it possible for my second school to implement a Bring Your Own Device (BYOD) policy. All but one student in my second group brought their own laptop to school, whereas this was not the case in my first group. Therefore, my first group of participants relied on the availability of the school's computer lab. This greatly limited when we could work with Scratch and for how long. To complicate things further, a number of my students received pull-out student services support during various times over the course of a day, thus making it difficult to select a time when all of my students in my class could participate in my study. In fact, there was only one time a week (Wednesday afternoons) when both the computer lab was available and most of my students were present. As a result, many of my students in my first group did not get the opportunity to finish their tutorials and start their own projects.

A challenge my first group of students faced was the inconsistency with how the Scratch interfaced was laid out to users. Depending on the Internet browser used, the step-by-step tutorial drawer would behave differently. Figure 27 shows a screenshot of the Scratch interface with the tutorial drawer opened. On certain browsers, the drawer would retract itself to the right-hand side of the screen as students follow the tutorial steps and drag blocks into the workspace, which allows students to have a bigger view of the workspace. However, on some browsers the drawer would disappear completely, and students would have to open it up manually. When they do, the drawer would open up to the main tutorial page. That meant students had to laboriously relocate not only the tutorial but also the specific step they were on — every time the drawer disappears. On the other hand, there were browsers that did not support the retracting of the tutorial drawer, and it ended up obscuring much of the workspace for students. Closing the drawer would again mean relocating the tutorial steps every time. To alleviate this problem, I have decided to have my second group of student follow similar tutorials that are presented in Portable Document Format (PDF). Though this eliminated the frustrations my students faced, these tutorials did not provide the same experience since the tutorial drawer features short video clips of the steps that otherwise are not available on a static document.

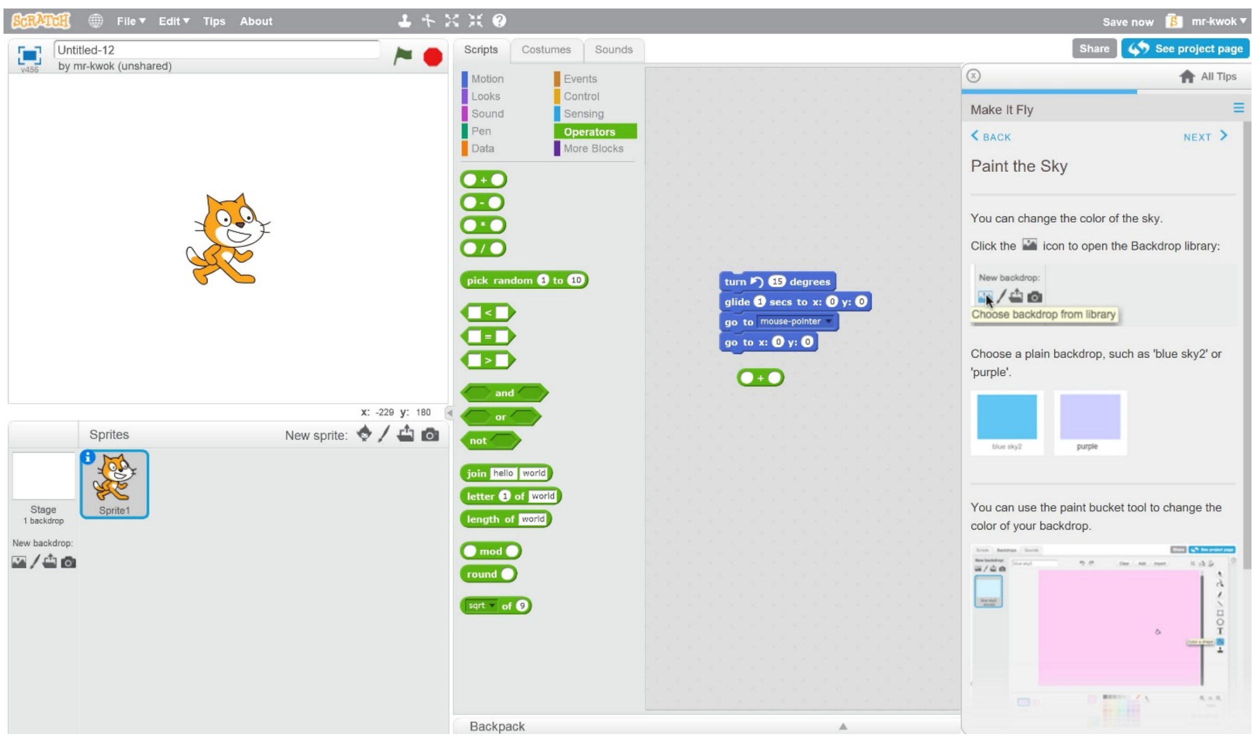


Figure 27. The Scratch interface with the Step-by-Step tutorial drawer pulled out. It takes up half of the workspace.

Thirdly, due to relocating to a new school in the middle of my inquiry, I went from teaching in a Montessori classroom to teaching in the neighbourhood program. Despite the fact that my second group of students have never been put into a formal Montessori learning environment before, I wanted to keep my research approach as close to the first group as possible. This meant being more of a guide-on-the-side and let my students explore and discover on their own than a sage-on-the-stage who gives direct instructions on how to use Scratch. The approach presented a different learning experience for my students, as I observed they often relied on me to show them the “correct” way of doing things in Scratch rather than spending time to figure things out independently. This might have affected how my students responded to my survey on their experience with using Scratch.

Another limitation to my study is the fact that some of my students’ parents did not give consent for me to collect data from their children. Despite my findings, my argument on using Scratch as a Montessori material to develop digital literacy would have been more powerful if my sample size were bigger. In my first group, 18 out of 22 students (82% participation rate) provided data for this study, while 26 out of 30 students (87% participation rate) in my second group completed my surveys.

Lastly, when I designed my interview and survey questions, the language or phrasing I used might have caused students to misunderstand what was being asked. For instance, when I listened to the recordings of my interviews, I noticed that my students sometimes did not know how to answer my questions, or their answers were unrelated to what I was asking them. Similarly, when reading the survey results given by my second group of students, a number of them provided irrelevant responses. For example, 35% did so on one of my questions as shown on Figure 18.

In all, despite having findings that support my hypothesis, there were a number of limitations that might have prevented me from presenting a stronger case. The next time I perform a teacher inquiry that is similar in nature, I will consider tweaking some of the research design and methodologies in order to provide a sturdier ground for my argument.

Future Recommendations

Aside from making adjustments to minimize limitations, there are a few recommendations to put forth for any future research. First, rather than teaching programming skills and computational thinking skills in parallel, it can be beneficial for students to learn about computational thinking first before doing any programming activities. The ability to decompose problems, identify inherent patterns, develop abstractions, and establish algorithms are valuable skills that could effectively help students in learning how to code and debug programs. A lot of my students became frustrated when they could not identify or locate bugs in their scripts when using Scratch. With sufficient computational thinking skills, I predict they would persist longer in trying to correct their errors rather than giving up quickly and starting over. Interestingly, one way to help students gain proficiency in computational thinking skills is to have them do activities without a computer. Games such as *Guess Who* (pattern matching, decomposition), *Sudoku* (abstraction), and solving a Rubik's Cube (algorithmic thinking) are activities that are often used as "unplugged" activities to promote concepts of Computer Science to young people. There are a plethora of resources such as *CS Unplugged* (http://csunplugged.org) that offer teachers ideas that could be used with students before introducing them to computer programming.

Next, if Scratch were to be used as a Montessori material to learn how to code, teachers could consider using similar program such as *Hopscotch* (Figure 28), *Lightbot* (Figure 29), or *ScratchJr* as a way to transition students into Scratch, especially in a classroom with younger students. Although they might not adhere to the Montessori principle of learning and teaching as comprehensively as Scratch, these programs offer a glimpse into the concept of block programming that underlies Scratch but using a much simpler interface and providing a more gradual learning curve.

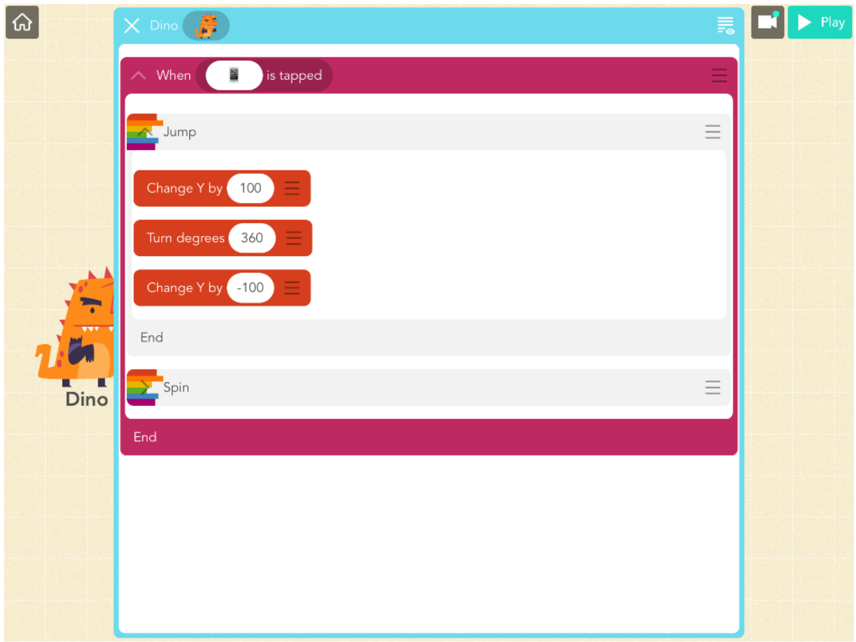


Figure 28. Block-based programming using an iPad app called Hopscotch.

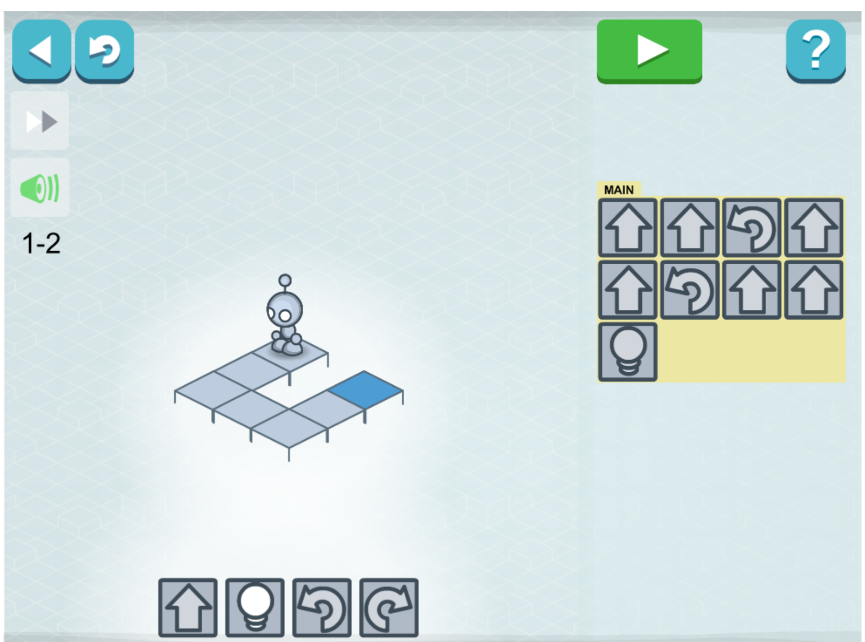


Figure 29. Another block-based programming iPad app called Lightbot.

Finally, moving forward, I hope my inquiry has inspired Montessori educators to not only consider using Scratch in their classrooms to help students develop digital literacy through computer programming, but to also perform their own inquiries into looking at other digital technologies that could be integrated into the Montessori learning environment. Moreover, I predict the next generation of students will continue to be exposed to an increasing amount and variety of digital media; in order to be truly digitally literate, these students must not be mere consumers of media but also producers to be successful citizens of the 21^{st} century and beyond.

References

Adams, A. H. (1970). Selected principles and methodology of Maria Montessori. *Educational Horizons, 48*(4), 124-128.

Akcaoglu, M. (2014). Learning problem-solving through making games at the game design and learning summer program. *Educational Technology Research and Development, 62*(5), 583-600.

American Montessori Society. (2015). FAQs. Retrieved from http://amshq.org/Montessori-Education/FAQs

BC Ministry of Education. (1996). *Information technology 8 to 10: Integrated resource package*

BC Ministry of Education. (2003). *Information and communications technology 11 and 12: Integrated resource package* (2004th ed.)

BC Ministry of Education. (2015a). Curriculum overview. Retrieved from https://curriculum.gov.bc.ca/curriculum/overview

BC Ministry of Education. (2015b). Curriculum redesign. Retrieved from https://curriculum.gov.bc.ca/rethinking-curriculum

BC Ministry of Education. (2015c). *Learning standards and flexible learning environments*

BC Ministry of Education. (2016). *Introduction to applied design, skills and technologies (draft).*

Berger, M. (2016, January 2016). Technology and Montessori: Thumbs up or thumbs down? *Tomorrow's Child, 24*, 9-12.

Beydler, J. A. (2009). Cosmic education: The heart of the Montessori elementary classroom. Retrieved from http://www.forsmallhands.com/ideas-insights/cosmic-education

Bobbitt, F. (1918). *The curriculum*. Cambridge, MA: The Riverside Press.

Boyd, B. F. (2008). Assistive technology for every child. *Montessori Life: A Publication of the American Montessori Society, 20*(1), 30-35.

Buckleitner, W. (2010). What would Montessori say about the iPad? *Children's Technology Review, 18*, 5-7.

C21 Canada. (2012). *Shifting minds: A 21st century vision of public education for Canada*. Toronto, ON: C21 Canada.

Chase, Z., & Laufenberg, D. (2011). Embracing the squishiness of digital literacy. *Journal of Adolescent & Adult Literacy, 54*(7), 535-537.

Chattin-McNichols, J. (1996). Technology in the Montessori classroom: The most frequent questions. *Montessori Life, 8*(1), 37-41.

Cloud Tools (n.d.). Retrieved from http://www.sd43.bc.ca/Resources/DigitalCitizenship/Pages/CloudTools.aspx

Cooney, A., & Jones, S. (2011). The educational theory of Maria Montessori. Retrieved from http://www.newfoundations.com/GALLERY/Montessori.html

Cossentino, J., & Brown, K. (2015). Assessing creativity and critical thinking in schools: Montessori as a holistic intervention. *Association Montessori Internationale Journal,* (2014-2015), 229-232.

Denner, J. (2011). Computer games created by middle school girls: Can they be used to measure understanding of computer science concepts? *Computers and Education, 58*(1), 240-249.

Dennis, P. A. (1974). Levi-strauss in the kindergarten: The Montessori pre-schooler as bricoleur. *International Review of Education / Internationale Zeitschrift Für Erziehungswissenschaft / Revue Internationale De L'Education, 20*(1), 3-16.

Duckworth, C. (2006). Teaching peace: A dialogue on the Montessori method. *Journal of Peace Education, 3*(1), 39-53.

Dunn, S. E. (2000). Technology: Where is it taking us? A "writing" perspective. *Montessori Life, 12*(1), 34-35.

Eshet-Alkalai, Y. (2004). Digital literacy: A conceptual framework for survival skills in the digital era. *Journal of Educational Multimedia and Hypermedia, 13*(1), 93-106.

Freire, P. (1970). *Pedagogy of the oppressed*. New York, NY: Continuum.

Goodwin, V. M. (2010). Virtual vs. reality: Montessori education and today's technology. *Children's Technology Review, 18*, 8.

Graham, A., Powell, M., Taylor, N., Anderson, D. & Fitzgerald, R. (2013). Ethical Research Involving Children. Florence: UNICEF Office of Research – Innocenti. Retrieved from: http://childethics.com/wp-content/uploads/2013/10/ERIC-compendium-approved-digital-web.pdf

Hardy, M. C. (1917). The derivation of the Montessori didactic apparatus. *The Elementary School Journal, 18*(4), 294-300.

Helfrich, M. S. (2011). *Montessori learning in the 21st century: A guide for parents & teachers*. Troutdale, Oregon: NewSage Press.

Herman, J. M. (2012). Creating balance in the new age of technology. *Montessori Life, 24*(3), 36-43.

Hubbell, E. R. (2003). Integrating technology into the Montessori elementary classroom. *Montessori Life, 15*(2), 40-41.

Hubbell, E. R. (2006). Authenticity and technology in Montessori education. *Montessori Life, 18*(2), 16-20.

Interagency Advisory Panel on Research Ethics. (2014). *Consent.* Retrieved from http://www.pre.ethics.gc.ca/eng/policy-politique/interpretations/consent-consentement/

International Society for Technology in Education. (2016). *ISTE standards for students 2016*. Arlington, VA: International Society for Technology in Education.

Israel, M., Wherfel, Q. M., Pearson, J., Shehab, S., & Tapia, T. (2015). Empowering K-12 students with disabilities to learn computational thinking and computer programming. *Teaching Exceptional Children, 48*(1), 45-53.

Kafai, Y. B., & Burke, Q. (2013). Computer programming goes back to school. *Phi Delta Kappan, 95*(1), 61-65.

Kafai, Y. B., Peppler, K. A., Lemke, J., & Warschauer, M. (2011). Youth, technology, and DIY: Developing participatory competencies in creative media production. *Review of Research in Education, 35*, 89-119.

Kahn, K., & Spiegel, H. L. (1999). The role of computer programming in education. *Journal of Educational Technology & Society, 2*(4), 6-9.

Kalelioglu, F., & Gülbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education, 13*(1), 33-50.

Krauss, J., & Prottsman, K. (2017). *Computational thinking and coding for every student*. Thousand Oaks, CA: Corwin.

Lankshear, C., & Knobel, M. (Eds.). (2008). *Digital literacies: Concepts, policies and practices*. New York, NY: Peter Lang Publishing Inc.

Lillard, A. S. (2005). *Montessori: The science behind the genius*. New York: Oxford University Press.

Lillard, A. S. (2008). How important are the Montessori materials? *Montessori Life, 20*(4), 20-25.

Lillard, A. S. (2013). Playful learning and Montessori education. *American Journal of Play, 5*(2), 157-186.

Lillard, P. P. (1972). *Montessori: A modern approach*. New York, NY: Schocken Books Inc.

Lillard, P. P. (1996). *Montessori today: A comprehensive approach to education from birth to adulthood*. New York, NY: Schocken Books Inc.

Love, A., & Sikorski, P. (2000). *Integrating technology in a Montessori classroom* (Guides – Non-Classroom). Washington, DC: U.S. Department of Education.

Marsh, C., & Willis, G. (2007). *Curriculum: Alternative approaches, ongoing issues* (4th edition ed.). Upper Saddle River, NJ: Pearson/Merrill Prentice Hall.

Media Awareness Network. (2010). *Digital literacy in Canada: From inclusion to transformation*. Ottawa, ON: Media Awareness Network.

Monroy-Hernández, A., & Resnick, M. (2008). Empowering kids to create and share programmable media. *Interactions, 15*(2), 50-53.

Montessori, M. (1912). *The Montessori method* (A. E. George Trans.). New York, NY: Frederick A. Stokes Company.

Montessori, M. (1917). *Spontaneous activity in education* (F. Simmonds, Trans.). New York, NY: Frederick A. Stokes Company.

Montessori, M. (1949). *The absorbent mind*. Adder, India: The Theosophical Publishing House. Retrieved from https://ia801404.us.archive.org/0/items/absorbentmind031961mbp/absorbentmind031961mbp.pdf

Montessori, M. (1949/2007). *The formation of man* (A. M. Joosten, Trans.). Amsterdam, NL: Montessori-Pierson Publishing Company.

Montminy, P. (1999). Best practice guidelines for computer technology in the Montessori early childhood classroom. *Montessori Life, 11*, 30-31.

Navarrete, C. C. (2013). Creative thinking in digital game design and development: A case study. *Computers & Education, 69*, 320-331. doi:http://dx.doi.org.ezproxy.library.ubc.ca/10.1016/j.compedu.2013.07.025

North America Montessori Center. (2009). The five great lessons for Montessori elementary: An introduction and lesson idea list. Retrieved from http://montessoritraining.blogspot.ca/2009/09/introducing-and-exploring-five-great.html

O'Reilly, T. (2005). What is web 2.0. Retrieved from http://www.oreilly.com/pub/a/web2/archive/what-is-web-20.html?page=1

P21. (2015). *Framework for 21st century learning.* Partnership for 21st Century Learning.

Peppler, K. A., & Kafai, Y. B. (2005). *Creative coding: Programming for personal expression.* Unpublished manuscript. Retrieved July 16, 2015, Retrieved from https://download.scratch.mit.edu/CreativeCoding.pdf

Peppler, K. A., & Kafai, Y. B. (2007). What videogame making can teach us about literacy and learning: Alternative pathways into participatory culture. Proceedings from DiGRA 2007: *Situated Play,* Tokyo, Japan.

Piaget, J. (1952). *The origins of intelligence in children.* (M. Cook, Trans.). New York, NY: International University Press, Inc.

Pierce, M. (2013). 21st century curriculum: Coding for middle schoolers. *T.H.E.Journal,* 40(5), 20-23.

Powell, M. (2009). Is Montessori ready for the Obama generation? *Montessori Life, 21*(2), 18-29.

Powell, M. (2015). Modernizing Montessori: Who are the digital natives? *Tomorrow's Child, 23*, 31-36.

Prensky, M. (2001). Digital natives, digital immigrants. *On the Horizon, 9*(5), 1-6.

Prozesky, K., & Cifuentes, L. (2014). The Montessori approach to integrating technology. *Issues in Early Education, 1*(24), 29-38.

Readhead, Z. (2008). Summerhill school. In M. Hern (Ed.), *Everywhere all the time: A new deschooling reader* (pp. 120-124). Oakland, CA: AK Press.

Resnick, M. (2012). Let's teach kids to code [video file]. Retrieved from http://www.ted.com/talks/mitch_resnick_let_s_teach_kids_to_code/

Robertson, J. (2012). Making games in the classroom: Benefits and gender concerns. *Computers & Education, 59*(2), 385-398. doi:http://dx.doi.org.ezproxy.library.ubc.ca/10.1016/j.compedu.2011.12.020

Schneider, C. (2012). 5 characteristics connecting Montessori ed & the digital learning movement. Retrieved from http://gettingsmart.com/2012/05/5-characteristics-connecting-Montessori-ed-the-digital-learning-movement/

ScratchJr - home. (2017). Retrieved from http://www.scratchjr.org/

Sihirli Bahçe Montessori School. (n.d.). Montessori materials. Retrieved from http://www.sihirlibahce.com.tr/en/Montessori/Montessori-araclari/

Silcoff, S. (2016, January 17). B.C. to add computer coding to school curriculum. *The Globe and Mail.*

Spector, M. J. (2013). Emerging educational technologies and research directions. *Journal of Educational Technology & Society, 16*(2), 21-30.

Su, A. Y. S., Huang, C. S. J., Yang, S. J. H., Ding, T. J., & Hsieh, Y. Z. (2015). Effects of annotations and homework on learning achievement: An empirical study of Scratch programming pedagogy. *Educational Technology & Society, 18*(4), 331-343.

Teitelbaum, K. (2008). Curriculum. In Mathison, Sandra & Ross, W. E. (Ed.), *Battleground school* (pp. 168-177). New York, NY: Greenwood Press.

Torrence, M. (2012). Montessori education: An idea whose time has come? *Montessori Life, 24*(2), 18-23.

Tosco, J. E. (2015). *The effects of technology on engagement and retention among upper elementary Montessori students* (Masters dissertation). Retrieved from SOFIA (Paper 122).

Webb, H. C. (2013). *Injecting computational thinking into computing activities for middle school girls* (Doctoral dissertation). Retrieved from UMI (UMI 3576592)

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33-35.

Woolsey, K., & Woolsey, M. (2008). Child's play. *Theory into Practice, 47*(2), 128-137.

Yang, Y. C., & Chang, C. (2013). Empowering students through digital game authorship: Enhancing concentration, critical thinking, and academic achievement. *Computers & Education, 68*, 334-344. doi:http://dx.doi.org.ezproxy.library.ubc.ca/10.1016/j.compedu.2013.05.023

Appendix A

Summary of Literatures Reviewed

Table 2. ***Summary of type and number of literatures reviewed***

Type of Literature	Number of Articles
Journal articles	28
Professional magazine articles	15
Books	7
Reports from government or non-profit organizations	6
Dissertations	4
Webpages	4
Conference proceedings	1
Total	65

Table 3. ***Sources on Montessori education that referenced 21st century competency as outlined by C21 Canada (2012)***

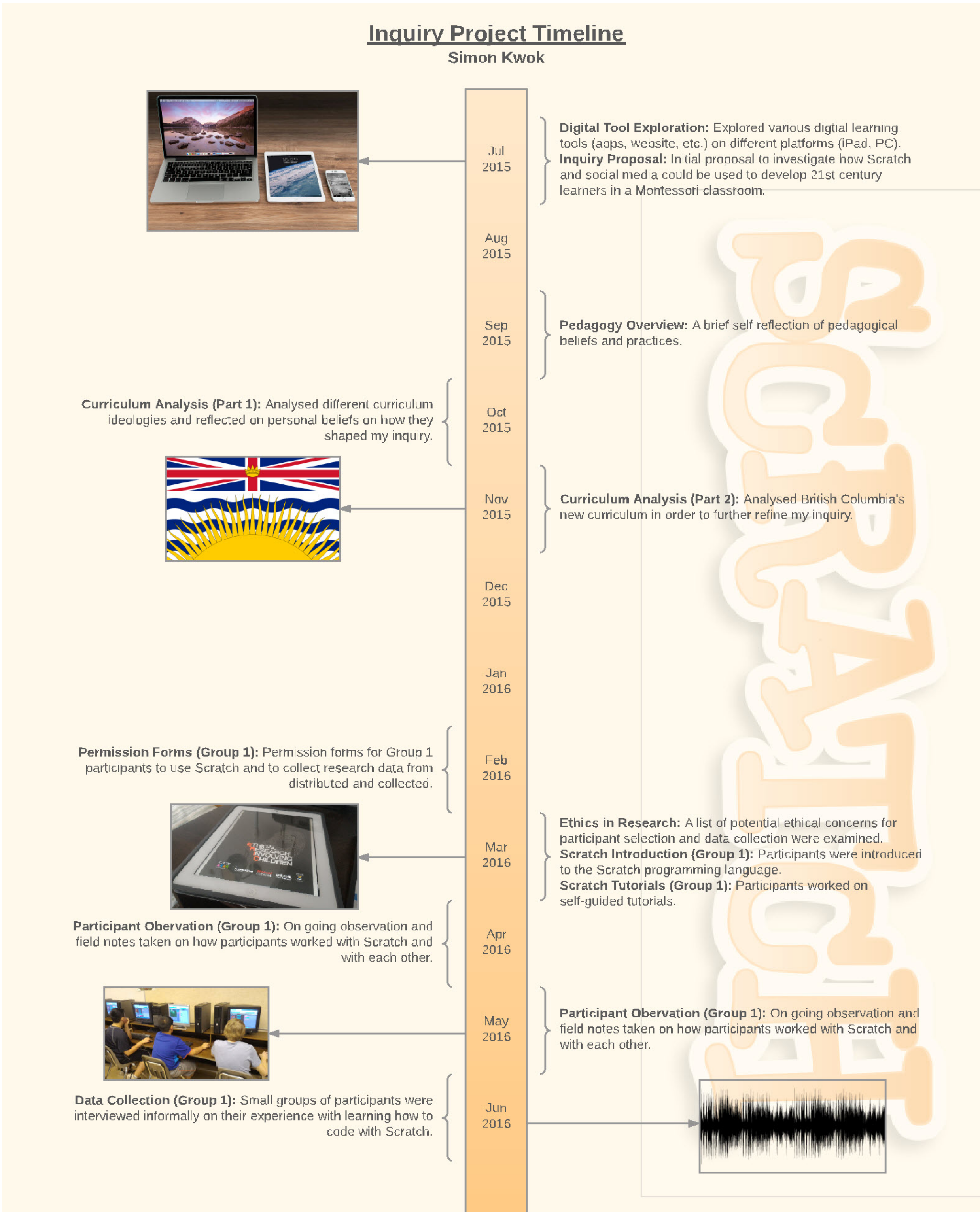
21st Century Competency	Sources
Character	Adams (1970) P.P. Lillard (1996) P.P. Lillard (2011) Montessori (1912) Montessori (1949) Torrence (2012)
Collaboration	Montessori (1912) Powell (2009) Torrence (2012)
Communication	Helfrich (2011) Montessori (1949)
Creativity	Cossentino & Brown (2015) A. S. Lillard (2011) Montessori (1912) Montessori (1917)
Critical Thinking	Cooney & Jones (2011)
Culture and Ethical Citizenship	Montessori (1912) Montessori (1949) Torrence (2012)

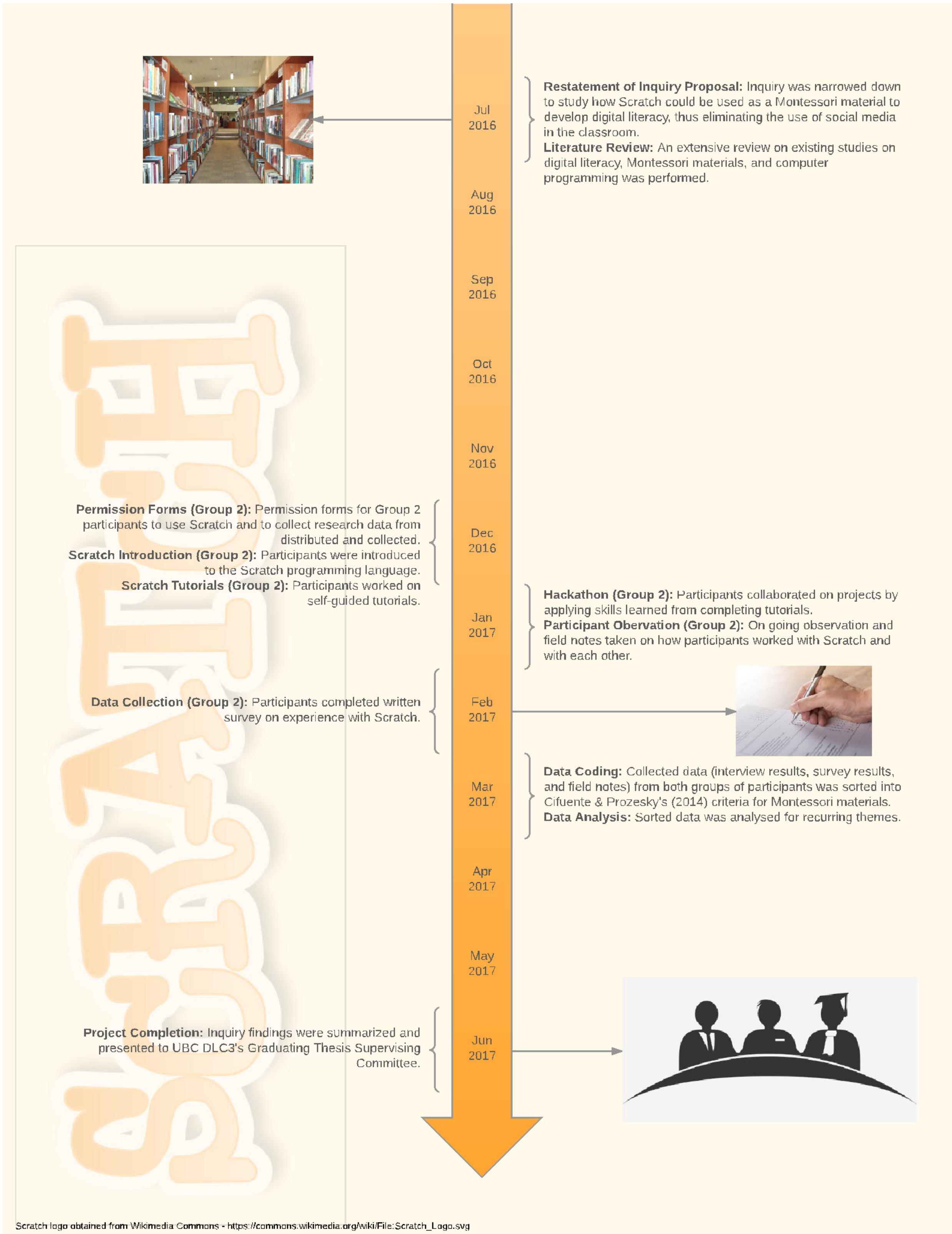
Table 4. ***Sources on computer programming that referenced the achievement indicators as outlined by the International Standards of Technology in Education (ISTE)***

Achievement Indicator	Sources
Empowered Learner	Akcaoglu (2014) Kahn & Spiegel (1999) Yang & Chang (2013)
Digital Citizen	Peppler & Kafai (2007)
Knowledge Constructor	Akcaoglu (2014)
Innovative Designer	Kahn & Spiegel (1999) Navarette (2013)
Computational Thinker	Denner (2011) Israel et al. (2015) Webb (2013) Wing (2006)
Creative Communicator	Denner (2011) Robertson (2012)
Global Collaborator	Israel et al. (2015)

Appendix B

Project Timeline




Jul 2016
Restatement of Inquiry Proposal: Inquiry was narrowed down to study how Scratch could be used as a Montessori material to develop digital literacy, thus eliminating the use of social media in the classroom.
Literature Review: An extensive review on existing studies on digital literacy, Montessori materials, and computer programming was performed.
Aug 2016
Sep 2016
Oct 2016
Nov 2016
Dec 2016
Permission Forms (Group 2): Permission forms for Group 2 participants to use Scratch and to collect research data from distributed and collected.
Scratch Introduction (Group 2): Participants were introduced to the Scratch programming language.
Scratch Tutorials (Group 2): Participants worked on self-guided tutorials.
Jan 2017
Hackathon (Group 2): Participants collaborated on projects by applying skills learned from completing tutorials.
Participant Obervation (Group 2): On going observation and field notes taken on how participants worked with Scratch and with each other.
Feb 2017
Data Collection (Group 2): Participants completed written survey on experience with Scratch.
Mar 2017
Data Coding: Collected data (interview results, survey results, and field notes) from both groups of participants was sorted into Cifuente & Prozesky's (2014) criteria for Montessori materials.
Data Analysis: Sorted data was analysed for recurring themes.
Apr 2017
May 2017
Jun 2017
Project Completion: Inquiry findings were summarized and presented to UBC DLC3's Graduating Thesis Supervising Committee.
SCRATCH
Scratch logo obtained from Wikimedia Commons - https://commons.wikimedia.org/wiki/File:Scratch_Logo.svg

Appendix C

Writers Workshop Planning Page to Gather Participant Input for Scratch

Example of a review planning page (Scratch)

Introduction (The Background): What is Scratch?

Criteria #1 (Graphics): Does the way the Scratch programming interface is designed engage people to use it to learn how to code? Why and/or why not?

Criteria #2 (Ease of use): Is there a steep learning curve for people to use Scratch to learn how to program and create their own projects? Why and/or why not?

Criteria #3 (Self correction): Is it easy for people to know when something is not working in Scratch? Is it easy for people to solve problems with their projects? Why and/or why not?

Criteria #4 (Creativity): Does Scratch allow people to be creative? Why and/or why not?

Criteria #5 (Collaboration): Does Scratch allow people to work together on projects? Why and/or why not?

Conclusion (The Summary):

Pros	Cons

The Bottom Line: A few summarizing points you leave the audience with.

Appendix D

Scratch Hackathon Self Evaluation Form

Name: ____________________________

Hackathon Self Evaluation

	Student Evaluation (1-4)	Evidence	Teacher Evaluation (1-4)
Defining: Identify criteria for success and any limitations			
Ideating: Generate potential ideas and add to others' ideas Screen ideas against criteria and limitations Choose an idea to pursue			
Prototyping: Identify and use resources to help create your product Develop a plan and identifies key stages and resources Constructs a first version of the product, making changes as needed Record your progress on OneNote (Group progress page and 3 check-in sheets)			
Testing Gather peer and/or user feedback and inspiration Troubleshoot bugs and make appropriate changes			

Name: ____________________________

What did you like about your shared product?

What could still be improved with your shared product?

What's a highlight from your experience throughout the Hackathon?

What were some frustrations that you had throughout the Hackathon? How did you manage these frustrations? Did you do so independently?

The BIG IDEA for this project is to understand that complex tasks such as creating something in the Hackathon, require acquisition of additional skills. Think back on your experience with using Scratch (Scratch Surprise, 10 Blocks, Tutorials, Hackathon), what kinds of skills did you think you have acquired or improved over the past month? *Be as specific as you can. Skills could be academic, personal, or social.*

Appendix E

UBC Letter of Invitation and Information and Consent Form

THE UNIVERSITY OF BRITISH COLUMBIA | VANCOUVEF

Department of Curriculum and Pedagogy

January 2017

LETTER OF INVITATION AND INFORMATION

Dear Parents of Students in Division 1,

Your child is invited to participate in a research study that is investigating the use of Scratch to develop digital literacy in a middle school classroom.

The aim of this letter is twofold. First, it describes the purpose and method of this inquiry. Second, it requests that you agree in writing that you will allow your child to participate in this teacher inquiry study. Please indicate your decision to allow your child to participate in this study on the attached consent form.

The study is entitled **Starting from Scratch: Using Scratch as a Montessori Material to Develop Digital Literacy** and is sponsored by the Faculty of Education at the University of British Columbia and will be conducted by Dr. Stephen Petrina. This research will be used for the Master's research project of Simon Kwok. With the introduction of the Applied Design, Skills, and Technology (ADST) curriculum by the BC Ministry of Education in which digital literacy is a part of, this study looked at how Scratch, a programming language created by the Massachusetts Institute of Technology, would help students develop this skill. Students will explore coding with Scratch, first going through self-guided tutorials then proceeding to design and create a program (game, animation, or app) of their choice. At the end of the project, participants provided their experience with using Scratch through informal small group conversations as well as a Writer's Workshop assignment on reviewing the programming tool.

Results of this research will be used in a graduate thesis and we intend to publish/share the findings of the study in professional journals and report or present them at conferences/professional development. At no time will the actual identity of the participants or their classroom be disclosed. Participants will be assigned pseudonyms and these will only be used in publication. We will maintain the strictest levels of protocols towards any and all information revealed. Participation is voluntary and you may choose to withdraw your child from the study anytime.

If you have any questions or desire further information with respect to the study you may contact Dr. Stephen Petrina at 604-822-5325 or email stephen.petrina@ubc.ca.

Sincerely,

Simon Kwok

THE UNIVERSITY OF BRITISH COLUMBIA | VANCOUVEF

Department of Curriculum and Pedagogy

Consent Form

Starting from Scratch: Using Scratch as a Montessori Material to Develop Digital Literacy

Investigator
The principal investigator for this study is Dr. Stephen Petrina, member of the Faculty of Education and who may be reached at (604) 822-5325. This research will be used for the M.Ed thesis of **Simon Kwok**, a graduate student in the Faculty of Education.

Study Purpose and Procedures
With the introduction of the Applied Design, Skills, and Technology (ADST) curriculum by the BC Ministry of Education in which digital literacy is a part of, this study looked at how Scratch, a programming language created by the Massachusetts Institute of Technology, would help students develop this skill. Students will explore coding with Scratch, first going through self-guided tutorials then proceeding to design and create a program (game, animation, or app) of their choice. At the end of the project, participants provided their experience with using Scratch through informal small group conversations as well as a Writer's Workshop assignment on reviewing the programming tool.

Confidentiality
Your identity will be kept strictly confidential. All documents will be identified only by code. Physical hard copies will be kept in a locked filing cabinet. Electronic copies will be encrypted and protected by password. This data will be kept in the vault in the office at the Port Moody Secondary School and will be accessed only by research team members.

Contact Information
If you have any questions or desire further information with respect to this study, you may contact Dr. Stephen Petrina at (604) 822-5325. If you have any concerns or complaints about your rights as a research participant and/or your experiences while participating in this study, contact the Research Participant Complaint Line in the UBC Office of Research Ethics at 604-822-8598 or if long distance e-mail RSIL@ors.ubc.ca or call toll free 1-877-822-8598.

Consent
Your participation in this study is entirely voluntary and you may refuse to participate or withdraw from the study at any time. Your signature below indicates that you have received a copy of this consent form for your own records. Your signature indicates that you consent to participate in this study.

Participant Signature		Date
Printed Name of the Participant		
Parent/Guardian Signature		
Printed Name of the Parent/Guardian		

Appendix F

District Permission Form for Scratch

Consent for Storage and Access of Information Outside Canada (Scratch)

In our class, students will be using **Scratch** (http://scratch.mit.edu/terms_of_use/) to support the ability to program their own interactive stories, games & animations and share their creations with others in the online community from school and home as part of classroom assignments. Students will be freely sharing code, art, music, and other works. It is the responsibility of the student to make sure they have the necessary rights, licenses, or permission for any user-generated content they submit to **Scratch**.

This information will be shared with:

☒ Public (for Shared Projects)
☒ Private (for Unshared Projects)

Students have been given teacher-created **Scratch** accounts to access the service, and their accounts will be deleted at the end of the term or school year. Note that **Scratch** is an online service located outside of Canada and is within the jurisdiction of the United States of America. Students should avoid storing materials that include information which could be used to identify themselves or other persons.

Kindly return a copy of this letter to **Mr. Kwok** signed and dated before **January 23, 2017**. If you have any questions or concerns, please feel free to contact **Mr. Kwok at** ________________.

Consent: I understand that the information my child may create and store will be stored in or accessed from a location outside of Canada and I hereby consent, on behalf of me and my child, to my child's information identified above being stored in, or accessed from, a location outside of Canada.

______________________ Signature of Parent or Guardian	______________________ Date
______________________ Print Name	______________________ Signature of Student (if over 13)
	______________________ Print Student Name and Grade
______________________ Print School Name	______________________ Print Teacher's Name