MEETING THE ‘T’ IN STEM THROUGH COMPUTER SCIENCE AND CODING

A graduate project submitted in partial fulfillment of the requirements for the degree of Master of Arts in Education, Elementary Education

By

Sara Castro

in collaboration with Maureen Straub

December 2015
The graduate project of Sara Castro is approved:

______________________________ ____________________
Raymond Brie, Ph. D. Date

______________________________ ____________________
Katherine Pilkinton, M.A. Date

______________________________ ____________________
Susan Belgrad, Ed. D., Chair Date

California State University, Northridge
ACKNOWLEDGEMENTS

I would like to thank my committee chairs, my family, friends, and colleagues. First off, I would like to thank Dr. Susan Belgrad for believing in STEM education and most of all, for believing in me. Without your knowledge and inspiration I would not have been able to become a STEM leader. You have taught me how important it is to be involved in STEM, as a teacher, student, and as a woman.

To Dr. Raymond Brie, you are an amazing educator with endless words of wisdom. Your ability to make math and technology fun and easy to integrate has made a great impact on my outlook on teaching. You taught me how to use technology as a teaching tool that both me and my students can enjoy.

To Katherine Pilkinton, you are a true example of someone who has dedicated her life to providing a nurturing and well-balanced education to students. Your leadership and support is the reason this project has come to life and touched the lives of so many students. I would not have had the determination to complete this project had it not been for the fact that I knew you would make it a reality at our school.

To my family and friends, I could not have realized this dream without your constant support and re-assurance. Without your encouragement I would not have been able to get through the long nights and weekends when I had stay home and do school work. Thank you for understanding and forgiving me when I couldn’t make it to special events or go on trips. You always let me know that there would be more to come. I appreciate you all for supporting me and giving me the time to reach my goal.

And finally, to Maureen Straub, one summer day in August you asked and I said yes!
Thank you for asking me to join you in this life-changing endeavor. Without your constant focus and shoulder nudging I would not have been able to get through it all. You never let me give up even when I thought I couldn’t do it anymore. You gave me pep talks, made me laugh, and made sure I was properly nourished. Thank you for being a great colleague, friend, and caretaker. We did it!
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature Page</td>
<td>ii</td>
</tr>
<tr>
<td>Acknowledgement</td>
<td>iii</td>
</tr>
<tr>
<td>Abstract</td>
<td>vi</td>
</tr>
<tr>
<td>Chapter One: Introduction</td>
<td>1</td>
</tr>
<tr>
<td>Technology-Enhanced Inquiry-Based Education in Elementary Education</td>
<td>1</td>
</tr>
<tr>
<td>Chapter Two: Literature Review</td>
<td>5</td>
</tr>
<tr>
<td>Closing the Computer Science Gap</td>
<td>6</td>
</tr>
<tr>
<td>Benefits of Learning to Code</td>
<td>8</td>
</tr>
<tr>
<td>Technology and Improved Content Knowledge</td>
<td>9</td>
</tr>
<tr>
<td>Technology and Motivation to Participate</td>
<td>11</td>
</tr>
<tr>
<td>Digital Natives and Technology</td>
<td>12</td>
</tr>
<tr>
<td>Chapter Three: Method of Development</td>
<td>15</td>
</tr>
<tr>
<td>About the Authors</td>
<td>15</td>
</tr>
<tr>
<td>About the T in STEM</td>
<td>16</td>
</tr>
<tr>
<td>Code.org Curriculum and Computer Science Week</td>
<td>17</td>
</tr>
<tr>
<td>About the Goal of the Resource Guide</td>
<td>18</td>
</tr>
<tr>
<td>Chapter Four: Webpage Resource Guide</td>
<td>19</td>
</tr>
<tr>
<td>Chapter Five: Discussion and Implications</td>
<td>31</td>
</tr>
<tr>
<td>References</td>
<td>33</td>
</tr>
<tr>
<td>Appendix: Addendum</td>
<td>37</td>
</tr>
</tbody>
</table>
MEETING THE ‘T’ IN STEM THROUGH COMPUTER SCIENCE AND CODING

By

Sara Castro

Master of Arts in Education, Elementary Education

Schools play a key role in closing the computer science gap that exists between student STEM achievement, college majors and career choices among U.S. students. Early exposure to computer science and coding may help increase the participation of underrepresented minority students (URM) and ensure that educators help our nation meet the demand for future computer scientists. The goal of this project is to provide an important, accessible resource guide that elementary school teachers can utilize to implement a school-wide plan that integrates computer science and coding in the curriculum to meet Next Generation Science Standards (NGSS).

Technology along with inquiry-based learning will continue to have strong impact on student motivation. Online, open source programs such as Genius Hour—an inquiry and technology-based learning model designed to improve student content knowledge and participation are already leading the way in understanding how as digital natives students can be meaningfully engaged with integrated-technology in the classroom. When implemented with knowledge and integrity, technology can serve as a key way to promote student achievement in
science, engineering and mathematics (STEM). This project responds to the need for introducing accessible, open source computer science and coding in early educational contexts that promotes student motivation especially when enhanced by inquiry-based models of instruction.

Keywords: Coding, Computer Science, STEM, Elementary Education, NGSS, Project Based Learning, 21st Century Skills.
CHAPTER ONE: INTRODUCTION

Technology-Enhanced Inquiry-Based Education in Elementary Education

STEM education (science, technology, engineering and mathematics) has come to the forefront in recent years as many high-tech companies have claimed that they cannot find enough qualified applicants to fill their technology jobs (Time, 2014). As demand for these jobs grow, so too do the salaries and quality of life of the workers who pursue STEM careers. According to the Bureau of Labor and Statistics (BLS, 2014), the mean annual wage of a computer programmer is $82,690 (http://www.bls.gov/oes/current/oes151131.htm). Computer systems design and related services has been projected by the BLS as one of the fastest-growing sectors. This demand should be met with a workforce capable of fulfilling those roles. Yet there has been little awareness among K-5 elementary school teachers that they are part of the “workforce development solution” (ACT, 2014) and that it is the responsibility of educators to assure that students leave secondary education with career-ready skills. Unfortunately, most of these educators remain unfamiliar with the nature of the workforce skills sought by STEM-sector employers, and are similarly unaware of the careers or “hot jobs in STEM” (U.S. Dept. of Labor, 2015) that their students can access when they are successful achievers in mathematics and science.

While K-5 teachers have recently adapted to the considerable shifts in pedagogy that are now expected in the Common Core State Standards (CCSS) (National Governors Association Center for Best Practices and Council of Chief State School Officers, 2010) they are unprepared to lead student-centered learning communities that assist students in achieving the “soft skills” demanded by today’s STEM employers: Communication, collaboration, critical thinking and
creativity (American Management Association and Partnership for 21st Century Skills, 2010). Advanced as the “4Cs” of the CCSS (Partnership for 21st Century Skills, 2010), teachers will soon realize that these shifts will be accelerated with the introduction of the Next Generation Science Standards (NGSS). Until teachers receive well-designed, engaging and motivating resources that address the intersections of these seismic shifts in education, misconceptions about the curricula, instruction, and materials required to teach the knowledge, skills and dispositions (KSD) needed by students will hinder both the teaching and the learning of these skills.

Once educators have been effectively led in becoming familiar with the materials and methods to teach computer science, (the T in STEM) they will begin to understand why the introduction of these concepts align with their students’ natural ability to use technology as both consumers and producers. Educators cannot assume that today’s students are all “digital natives,” as was originally described by Marc Prensky (2001). Critics such as Koutropoulos (2011) remind us that socioeconomic status is a factor in regards to labeling all students as “digital natives.” In a study conducted by Pew Internet and American Life Project (Rainie, 2011) 61% of digital natives indicated they are white and 54% claim to live in the suburbs. This statistic reminds us that overall, minority students have less access to digital technology.

This is especially important for teachers who work with diverse, students from underrepresented minorities (URM) because as Gee (2008) has argued, along with the “fourth-grade slump” in literacy, there is also a digital gap that is ever-widening for these students. “The fourth-grade slump consistently leads to educational failure. This “digital gap leads to a failure to become confidently ‘tech-savvy,’ a 21st-century skill crucial for success, and even for survival” (Gee, p. 2). A second and equally-important problem to be addressed by educators is
that the technology and computer science industry has been typically dominated by affluent white males. This lack of diversity not only reduces the number of qualified applicants for these lucrative workforce positions and careers, but also hinders innovation that arises when the workforce would otherwise be representative of our nation’s rich ethnic, racial, and gender tapestry.

According to the Computer Science Teachers Association, it is now more evident than ever before that a solid foundation in both the principals and practices of computer science are required if we expect that today’s students will grow up “to be well educated and to lead empowered lives [for which] our students will require a sound foundation in the principals and practices of computer science. There is no time too soon to begin the process of building that foundation” (Phillips, 2012, Para 1). For educators who teach in minority-serving elementary schools this is daunting but not insurmountable problem.

While it is true that by the time students have reached upper-elementary school the “fourth-grade slump” has occurred in literacy, it is similarly the beginning of a mathematics and digital gap (California STEM Learning Network, 2015). Each of these phenomena can be effectively addressed by teachers who have been provided with effective STEM KSD and access to motivating and ongoing resources that support them, such as those advanced by the California STEM Learning Network Educators. This organization asserts that the “T” in STEM can work together with other STEM disciplines to close the computer science gap and increase the advancement in STEM careers of women and URM students. The use of technology by K-12 students has been rapidly increasing in the world around us. According to the editor of the Computer Science Teachers Association publication, “Over the past few decades, computers
have transformed both the world our students live in and the world of work and innovation that they will join in the future” (Phillips, 2012, para 1). In the upper-elementary school classroom technology can increase productivity of teachers and students. Television, Internet, cellular phones, tablets, computers and digital devices are now a regular occurrence in our daily lives. Technology can be used in a variety of ways to support student learning in the classroom and for this reason increased use of technology in classrooms around the world has become an integral part of learning. Because “technology can be intrinsically motivating” to students, it is a great way to expand the learning that occurs within a classroom (Stipek, 1993). After all, when students are self- motivated, they often take pride in their work and celebrate their achievements.

It seems that using technology for purposes such as writing, research, and analysis can enhance student content knowledge as well as motivate them to participate in their own learning. Students are making connections with the technology they use every day at home and what they are learning at school. The desired goal is what matters, how they get there is up to them. By teaching them methods, procedures, and ways to find solutions with technology students can translate their own learning solutions. Using technology as a tool and not as a privilege is allowing students full access to endless learning. Researchers have found that students are “more engaged in learning” and attendance improves because of this motivation (Shapley, Sheehan, Maloney, & Caranikas-Walker, 2011).

Today’s students were born into a digital world, and use technology on a daily basis, both inside and outside of school. By an early age they have acquired their own personal devices that they carry around with them wherever they go. Easy access to these devices brings them to the World Wide Web with the stroke of their fingertips.
CHAPTER TWO: LITERATURE REVIEW

America may be the birthplace of computer science giants, such as IBM, Apple, Microsoft, and Google, but unfortunately over the last two decades it has fallen behind when it comes to training a diverse workforce of future computer scientists. According to the National Center for Education Statistics, computer science has seen a decrease from 25% to 19% of high school student participation over the last 20 years (NCES, 2009, p. 49). The National Science foundation (NSF) Science and Engineering Indicators 2012 reports that American women pursuing bachelor’s degrees in computer science has declined by 10% over the last 10 years.

By 2020, U.S. universities will not be able to fill even a third of the country’s 1.4 million computer science positions with qualified graduates. The industry needs to tap into the other 50% of the population if it hopes to find candidates for crucial jobs. At present, only 12% of computer science degrees go to women (Time, 2014).

Douglas Rushkoff, author of Program or Be Programmed: Ten commands for a digital age, suggests that America needs to incorporate computer programming into the core curriculum (Rushkoff, 2010). Hadi Partovi, an entrepreneur, investor and co-founder of Code.org, believes coding is a foundational skill for 21st century learners. In addition, Partovi (2014) claims that "To make computer science opportunities accessible to all students, we need to start in elementary school — where classrooms are split equally with students of all backgrounds, and the playing field is still relatively level" (New York Times, 2014, para 3).

As elementary school teachers that have implemented beginning coding skills into their instruction with a diverse population of students, the authors have experienced their students’ desire, motivation and excitement in learning computer science. At their Title 1 elementary
school they find it is the perfect place to implement a computer science education program. A one-to-one deployment of IPads will take place at this school during the 2015-2016 academic year. The authors like many other teachers who recognized the timeliness of school-based digital learning understands that there is no better time than now for the implementation of a computer science curriculum.

Their inspiration and interest in this research project to implement a computer science program began with the school’s participation in the nation-wide Hour of Code. To promote Computer Science Week (December 9 - 15, 2014), President Barack Obama participated in a world-wide campaign known as the “Hour of Code 2014”. He wrote his first line of code using an introduction to JavaScript with drag-and-drop tools devised by Code.org. The Obama Administration, the education community, businesses, foundations, and non-profit organizations have all recognized the need to help support and expand student access to computer science education in K-12 schools. After initiating and implementing school-wide participation of the Hour of Code at our school, our principal embraced the idea of incorporating computer science into our school curriculum. We have become intentional on creating a resource that enables K-8 educators to take advantage of similar opportunities.

Closing the Computer Science Gap

Research suggests that students exposed to coding skills and STEM (science, technology, engineering, and mathematics) curriculum at a young age clearly show the existence of less gender-based stereotype threat with respect to STEM careers (Metz, 2007; Steele, 1997). Schools play a key role in closing the computer science gap. Introductory concepts of computing, including beginning computer programming or coding, should begin as early as pre-
school and continue throughout the grade levels. Since our school is a STEM magnet, students beginning in kindergarten should be taught a computer science curriculum, such as the one offered by Code.org Partnerships, as well as those most currently established with Code.org, which have collaborated to provide: professional development for teachers, nationally-recognized courses and curriculum in a blended-learning environment, materials to promote computer science to parents and students, and support for all grade levels K-12.

In February, we completed Code.org’s seven-hour workshop that provided us with an entire curriculum and set of materials designed to bring computer science into the elementary school classroom through coding. This training is open to any and all K-5 teachers from the Greater Los Angeles area. We proposed and secured agreement from our principal that all the teachers at our school be trained to implement coding and computer programming.

There are countless online resources available, but one essential component is access. For most school systems, getting access to the Internet will require help from the federal government and wealthy philanthropic commitments from the likes of Google, and Microsoft. The Obama administration also intends to help close the computer-science education gap by expanding computer science education in schools. The National Science Foundation (NSF) plays an important part in advancing computer science education. The NSF has set long-term goals in the establishment of computer-science course availability to all schools.

Fortunately, the Los Angeles Unified School District (LAUSD) has access to the Internet on a regular basis. Our school in particular has excellent Internet accessibility. Closing the computer science gap at our school has begun by tapping into available resources beginning in
kindergarten. The benefits of learning computer science and coding for students of all ages has been recognized for many years.

**Benefits of Learning to Code**

Seymour Papert, the visionary former MIT professor, inventor of the Logo programming language, and author of *Mindstorms: Children, Computers, and Powerful Ideas* (Papert, 1980), has pioneered the idea that learning to code not only helps children better understand educational subject matter, but also teaches them how to think. More recently, Kafai and Burke, authors of *Connected Code: Why Children Need to Learn Programming* (*MIT, 2014*), reexamine Papert’s early movement of “learn to code,” and assert a similar view that teaching young children programming deepens computational thinking skills and thinking across subject matter as well as communicating and computational participation. Kazakoff (2012) and colleagues have found that learning to code in early childhood has shown positive effects on students’ sequencing abilities (Kazakoff & Bers, 2012; Kazakoff, Sullivan, Bers, 2013). In addition, their work has shown that young students are capable of understanding computational thinking, such as sequencing, control flow, loops, and branches. (Bers, 2008; Bers, 2010a; Bers & Horn, 2010; Bers et. al., 2002). Research by Douglas and Gullo (1984) concluded that first graders’ reflectivity and divergent-thinking scores showed significantly-higher measures, as well as an increase in some aspects of problem-solving ability after participating in a twelve-week computer programming course. Earlier, the Salomon and Perkins’ (1987) investigations of the impact of programming instruction on cognitive skills found that these can show occasional positive results under the right conditions. Wing (2006) has argued that computational thinking, or thinking like a computer scientist should be taught along with reading, writing, and
mathematics. Kazakoff (2014) states that, “learning to code also affords children an additional understanding of their digital worlds through learning the basic digital building blocks of the technologies around them” (p. 6).

**Technology and Improved Content Knowledge**

The ability to collaborate is important for success in the 21st century. The demand for critical thinkers, creative problem solvers and effective communicators is high in all career fields. Teaching students how to gain and use these skills is also important in today’s global economy. In the same way, teaching students through the use of technology can be intrinsically motivating to students. It is an important way to expand the learning that occurs within and outside of our classroom walls. Using technology for purposes such as writing, research and analysis can enhance student content knowledge, as well as motivate them to actively and responsibly participate in their own learning. Furthermore, using technology to teach can allow students to access content at their own pace.

It is important to remember that various technologies deliver different kinds of content and serve different purposes in the classroom. “Research has shown that the manner in which technology programs are implemented is equally, if not more, important than the type of technology used in the classroom” (Blazer, 2008, p. 3). We have personally seen an increase in students being able to demonstrate content knowledge through the use of, or with help from, integrated technology in my own teaching practice. Students have used the computer lab on the school campus on a regular basis, but most recently we began integrating the additional use of Ipads into our weekly repertoire of strategies. “Studies that isolate a particular technology to study its effect on some measure of student achievement find that well-implemented technology
use can lead to improved student achievement” (Grainger, 2006, para 1). In both of our classroom settings when students are exploring an educational game, taking a live online quiz on KAHOOT or NEARPOD, or challenging each other with self-selected tasks, they are intrinsically motivated to show evidence of what they know. We have found that using smartphones, tablets, and computers in our classrooms has been highly effective. As cited in learnnc.com, these tools can make a world of difference to students who may be struggling or uninterested in the disciplines taught in school. We find that the use of online flashcards to support students’ vocabulary development ensures that they are engaged and willing to learn the critical vocabulary terms. It is truly amazing how a task as simple as looking up definitions on a smartphone or tablet can become an important learning scaffold for a vocabulary lesson as it keeps students engaged and learning. Using the read function on a tablet is another great option for English Language Learners (ELLs) or any student who may struggle with reading or pronunciation, as the text is read aloud by the device. Being able to track individual progress, make and reach goals, and personalize learning for students can lead to improved student content knowledge. Therefore, technology can improve student content knowledge by serving as a scaffold to supplement learning.

As educators, we can use technology to help students learn the higher-order cognitive skills of analysis, synthesis, and evaluation. This will inevitably help to prepare them for their future education and careers. With technology-supported instruction we can give students opportunities to interact with one another as we encourage collaboration, both important aspects of being prepared for what their real-world environment is today. Inevitably, technology-
enhanced environments provide students with what seems almost limitless ways to explore and access information to support their ideas.

**Technology and Motivation to Participate in Learning**

A lack of motivation has important consequences. Former U.S. Secretary of Education, Terrell H. Bell said, “There are three things to remember about education. The first one is motivation. The second one is motivation. The third one is motivation” (Center on Education Policy, 2012, para 8). Any teacher knows that it is hard to keep students motivated all day and every day. Teachers have to keep up with the times in order to make sure that they are teaching meaningful content. A review of literature currently reveals that student motivation is connected to a teacher’s willingness to incorporate technology in everyday learning tasks. Motivation can be intrinsic or extrinsic, teachers need to encourage students and get them going (Sanacore, 2008). This is definitely evident in our students this year. We have had to modify assignments, increase the amount of time students are allowed to use technology, as well as how they can integrate it into their final products in order to keep them interested and responsibly participating.

Research shows that making learning active is a great way to ensure that students are engaged. What better way to engage students than through project based learning (PBL)? With PBL students can learn science content regardless of their language knowledge. Campbell (2012) studied how a PBL learning environment affects language learners as well as what the benefits and disadvantages might be. The results showed that students, both ELL and non- ELL, enjoyed the project-based learning environment. Furthermore, they noticed that collaboration allowed the
ELL students many opportunities for authentic academic talk time. They were more engaged and they were excited to learn.

Student motivation has for some time been described as one of the foremost problems in education. It is certainly one of the problems most commonly cited by teachers. Motivation is important because it “contributes to achievement”, but it is also important to consider the outcomes (Ames, 1984). With this in mind, we have opted to find ways to increase student motivation to gain content knowledge and to increase participation in our classrooms. We have found that increasing the use of technology, implementing more inquiry-based learning projects, and offering students opportunities to choose what or how they will learn the necessary content has made a positive impact on our student’s learning outcomes.

**Digital Natives and Technology**

Author Marc Prensky coined the term “digital natives” in 2001. Today’s students – K through college – represent the first generations to grow up with this new technology. They have spent their entire lives surrounded by and using computers, video games, digital music players, video cams, cell phones, and all the other toys and tools of the digital age (Prensky, 2001). Teachers need to keep this in mind when they plan and teach. Gone are the days when the only resources available to teachers are books and encyclopedias.

Children's traditional classroom tools - pencils, notebooks, and texts - are still vital. But technology applications allow students to study core subjects in a new way. They can take a new approach to understand the necessary content. Technology can be integrated into multi-curricular learning. Skills like critical thinking, problem solving, and collaboration are all included in learning with technology (Hill, 1996). Students in our classrooms love to use
technology, so they are sure to be interested in the content learning if it incorporates the tools they love to use. Integrating technology helps students stay engaged, and motivated to participate in their learning. We have seen a positive change in student attitudes toward learning since we started taking a more technology-rich approach to teaching.

As “digital natives” today’s students are accustomed to having technology at their fingertips, we have succumbed to the fact that it is simply more productive and often more efficient to allow students to use their own mobile devices, cell phones, and tablets to retrieve information from the World Wide Web, or take photographs for projects to help them create presentations. It takes only a few seconds to get a quick response to a possible research question or a correct spelling, synonym, antonym, or definition to a word. Students are highly familiar with the technology they use on a daily basis. Because of this, they are comfortable using it and they are motivated to use it as often as possible. If we can manage as a teacher, to make learning more interesting and exciting for them by integrating the use of technology, we must definitely try. Research shows that students in the 21st century are used to getting information quickly through the use of the Internet. In a journal that focused on middle school learners and technology the authors suggest that students look to Google or other search engines when they need answers. With YouTube offering free videos, the sky is the limit (Downes & Bishop, 2012). Twenty-first century thinkers learn and share through the use of technology. As educators, we can either go with or against the grain. It only makes sense to keep students actively learning and engaged. Technology is the tool to use for this.

The ultimate goals of this project are to inspire, train, and produce future computer scientists and knowledgeable 21st century learners. With the implementation of this project, we
hope to narrow the computer-science gap and digital divide beginning in the early elementary school years. The resource guide will be informed by the studies cited in the literature review as well as the success of a school-wide professional development led by the authors. The resource guide is created to help facilitate a school-wide plan to implement Code.org’s K-5 Computer Science Course 1, 2, and 3 with supplementary teacher support for the K-6 grade levels. Successful implementation of the resource guide should show evidence that students will be completing 20 minutes of coding a week either in the computer lab, in class, or online at home and teachers will be completing a one-day training and in order to be able to access online coding activities will register their class on http://studio.code.org/

In the next chapter, a detailed description of the content and how the resource guide found in the project is organized.
CHAPTER THREE: METHOD OF DEVELOPMENT

Meeting the ‘T’ In Stem through Computer Science and Coding: A Resource Guide

About the Authors

Maureen Straub and Sara Castro have conducted a joint project that has introduced, provided resources and demonstrated ways to integrate computer science and coding to all the teachers at their elementary school. Straub is a first-grade teacher and Sara Castro is a sixth-grade teacher. Both are members of their school’s technology team. The K-6 elementary school they currently work at is in their 2nd year as a STEMM (Science, Technology, Engineering, Math, Medicine) Magnet. Both completed Endeavor’s Coding, Robotics, and 1:1 Devices course, and seek to share this information with the teachers and staff at their school. With the deployment of 1:1 iPads for the 2015-16 school year, there is no better time than now to introduce and provide computer science and coding knowledge and resources to all 17 teachers and 2 administrators at their school. In addition, the school has purchased Project Lead the Way (PLTW) Launch for all grade levels. Straub has been trained as a Lead Teacher for PLTW Launch, and Castro will be trained as a lead teacher for PLTW Gateway. With the authors’ familiarity with NASA resources, PLTW, Endeavor courses, and affiliation with Code.org, it is expected that the Resource will be effective in placing a spotlight on the critical need for integration of computer science at elementary-level schools.
**About the T in STEM**

The ability to use technology is an important 21st century skill. Current statistics show that the need for computer scientists is high, yet the number of students who are technology proficient and prepared to thrive in the competitive work force of the 21st century is low (Time, 2014). Teaching students how to safely use technology will provide them with meaningful learning experiences and lifelong tools they can rely on in the future for college and careers. Starting students with coding at an early age will prepare our students to thrive and be competitive in the work force when the time comes.

According to research conducted by Margolis et al, knowledge of computer science has been exclusionary (2008). By introducing our students to computer science in grades K-6 we can provide them with a fundamental understanding of its principals. We hope to narrow the computer-science gap and digital divide beginning in the early elementary school years by doing our part to close the gap and make a difference in the lives of our underrepresented students.

Learning computer science and coding has benefits beyond a career in the computer science field. Coding skills can teach students how to think and how to understand sequencing, problem solving and critical thinking skills. Research has shown that the demand for critical thinkers, creative problem solvers and effective communicators is high in all career fields. Teaching students through the use of technology aides in the task of teaching content as well as providing them with equal access to resources they need to be successful throughout their educational careers. Furthermore, student motivation is increased through the use of technology integration. Recent research shows that making learning active is a great way to ensure student engagement and using technology as a learning tool is just the way to motivate young
learners. We have had great success thus far, with the implementation of coding in grades K-6, our students are motivated to move through the courses and learn all they can about computer programming, coding and computational thinking.

**Code.org Curriculum and Computer Science Week**

Code.org’s K-5 curriculum consists of three courses. The course structure was designed as a spiraling curriculum of computer science. The course is a blend of online activities and “unplugged” activities, lessons where students can learn computing concepts and skills with or without computers. The online activities consist of self-guided and self-paced tutorials. The online lessons use scaffolded sets of programming instructions that explore and practice algorithmic thinking. Some of the “unplugged” lessons given by the instructor contain hands-on, kinesthetic activities, and the use of physical manipulatives to model computational concepts. There are 17 lessons, each lasting 25 to 45 minutes. They can be taught in consecutive days or once a week. The content of each lesson builds upon the other. Code.org’s K-5 curriculum aligns to Computer Science Teachers Association (CSTA) Computer Science Standards and International Society for Technology in Education (ISTE) standards, while some lessons integrate national math, English language arts, and science standards.

Teachers will create a teacher account on Code Studio (http://studio.code.org), set-up a teacher dashboard and register their students as part of the one-day workshop. Teachers will be given a printed curriculum guide and classroom supplies for the unplugged computer science lessons that introduce vocabulary, computer science concepts, digital citizenship, perseverance and teamwork. Furthermore, teachers, students and parents will be given access to a webpage resource guide on computer science.
In addition, participation in Computer Science Week beginning December 7-13, 2015, will be implemented and scheduled into the school-wide events calendar. Throughout the 2015-2016 school year students will be able to code using their school iPads or the computers in the computer lab. The goal of coding for 20 minutes a week will be established throughout the school year at all grade levels.

**About the Goal of the Resource Guide**

The goal of the resource guide is to not only provide easy access to resources about computer science and coding for students, teachers and parents at the local STEM elementary school, but also to inspire other K-6 educators at schools across our nation to embrace the integration of computer science and to begin to examine the value of coding in an early educational context. All students, teachers, and parents will have access to the webpage resource guide through School Loop. A detailed description of the School Loop webpage (http://www.lfe-lausd-ca.schoolloop.com/coding) created by the authors will be described in the follow chapter.
CHAPTER FOUR: WEBPAGE RESOURCE GUIDE

Meeting the ‘T’ In Stem through Computer Science and Coding: A Resource Guide

A website resource guide on computer science provides an overview of the importance of teaching computer programming and learning to code by featuring teacher, parent, and student tabs with quick links, resources, videos and other important information as well as easy accessibility. Furthermore, evidence based on teacher surveys after the first professional development indicated the need of an accessible website resource guide.

The Los Feliz Magnet School STEMM coding website can be accessed via http://www.lfe-lausd-ca.schoolloop.com/coding as well as by clicking the link titled Coding: Meeting the “T” in STEMM tab from http://www.lfe-lausd-ca.schoolloop.com/.

Located at the top center of the webpage, the user will find images of posters provided by Code.org, which feature quotes by famous people promoting Computer Science Education Week and the importance of learning how to code. Below the slide show of posters is a link to Computer Science Education Week resources https://csedweek.org/. This site promotes the Hour of Code with a video and resources to sign up for the event being held December 7-13, 2015. In addition, a video promoting the new Stars Wars puzzles titled Building a Galaxy with Code, which was especially made for The Hour of Code can be viewed from the webpage.
While scrolling down the webpage toward the center a slide-show entitled “Why Computer Science?” reveals five different slides provided by Code.org addressing “The Job/Student Gap in STEM;” “1,000,000 Unfilled Jobs by 2020;” statistics on high school advanced placement (A.P.) enrollment in computer science coursework, the decreasing percentages of women computer science majors, and images of the effects of technology.
WHY COMPUTER SCIENCE?

Technology affects every field

The Job/Student Gap in STEM

- 36% of men & sciences
- 2% of women

Students

Jobs

60% computing jobs

40% engineering & sciences

Source: U.S. Dept. of Labor, National Science Foundation

WHY COMPUTER SCIENCE?

1,000,000 Unfilled Jobs by 2020

1,000,000 more jobs than students by 2020

$500 billion opportunity

1.4 million computing jobs

400,000 computer science students

Source: RSA, Inc., Employment Research Institute

WHY COMPUTER SCIENCE?

Fewer CS majors than 10 years ago (and a shrinking % are women)

Exposure to CS leads to the best-paying jobs in the world. But 75% of our population is underrepresented.

Explores 2012 High School A.P. Enrollment

2012 High School A.P. Courses

Less than 1000

In 2012, fewer than 1000 A.P. students graduated and required students took the high school for computer science exam.
To the right of the webpage a slideshow titled *Students Coding at Los Feliz STEMM* shows photographs of students engaged in both online and “unplugged” activities in the computer lab, on iPads and in the classroom. In addition, there is a photograph of a bulletin board displaying students certificates of completion of Course 1 from Code.org’s K-5 curriculum as well as student writing and drawings of things they learned about computer science using the Code.org curriculum.
Below the student photographs are quick-links to each teachers’ Uniform Resource Locator (URL) Studio.Code.org classroom section. These links are organized by grade level and provide easy access for students in the computer lab, on iPads in the classroom and/or outside of school online.
To the far left of the webpage underneath a short purpose statement about the intent of the webpage is a photograph of the teachers, administrators, and community participants in attendance for the one-day Professional Development Workshop co-facilitated by the authors and Dominic Caguioa, Affiliated Trainer for Code.org’s Introduction to K-5 Computer Science Courses.
In the Teacher Resources section, the following links are provided to help teachers make a connection with why we are coding. Articles, lessons, common core correlations, etc. are provided here. The first link https://code.org/educate/districts opens to a page titled, *Partnership with Code.org to bring K-12 computer science to your district*. This page provides educators with information about district partnerships, curriculum, and professional development. The following link http://csunplugged.org opens to a page titled, *CS Unplugged Computer Science without a computer*. This link provides a free book, videos and activities that teach computer science concepts such as algorithms, binary numbers and data compression. These activities introduce students to Computational Thinking with materials such as paper, string and crayons instead of an actual computer. The next resourceful link goes to http://www.jpl.nasa.gov/edu/, the education department of the Jet Propulsion Laboratory (JPL), which offers opportunities for teaching, learning and current events.
To answer the question about how computer science and the Common Core correlate an article from Edutopia titled, *Coding in the Common Core* can be retrieved from http://www.edutopia.org/blog/coding-in-the-common-core-tara-linney.

This article discusses the differences between programming and coding, Common Core correlations, integration ideas, and best practices when teaching computer programming.

The last two links in the Teacher Resources section of the resource guide webpage provide extensive information and resources about Scratch, an educational programming language developed by the Lifelong Kindergarten Group at the Massachusetts Institute of Technology (MIT). The first website titled, *Computational Thinking with Scratch: Developing Fluency with Computational Concepts, Practices, and Perspectives* located at http://scratched.gse.harvard.edu/ct/, answers the following questions: What is computational thinking (CT)? How do I assess the development of CT? How do I support the development of CT? The second website titled *ScratchEd* at http://scratched.gse.harvard.edu provides an online community for Scratch educators where they can find out more about Scratch, share stories, exchange resources, and ask questions.

Underneath the Teacher Resources section is the Parent Resources section. In this section, three different links are provided to help parents understand the importance of coding and how they can get involved. The first is the parent information page of Scratch located at https://scratch.mit.edu/parents/. A video of Mitch Resnick, the Director of the Lifelong Kindergarten group at MIT Media Lab, explains his reasons for teaching kids to code can be view on this page. In addition, parents can find out more by reading frequently asked questions,
their privacy policy and other helpful resources and information. The next link provides an overview and resources for Tynker, a visual programming language similar to Scratch. With Tynker located at https://www.tynker.com/ students can build web apps, custom games, art, and science projects. The third link, http://www.macworld.com/article/2922335/teach-your-kids-and-yourself-how-to-code-with-these-ipad-apps.html, is an online article that discusses how parents can teach themselves and their kids how to code by learning about a variety of applications (apps) such as Hopscotch, Tynker, Move the Turtle, Cargo-Bot, and Daisy the Dinosaur for coding on digital devices such as iPads.

The following sections of the webpage resource guide include quick links to Scratch Jr., Scratch, and Hopscotch. Included here are a variety of “Cool Coding Websites and Apps” featured at the bottom of the webpage, which also includes a link to a Common Sense Media websites that rates each one.
In the section “NASA and Computer Science” there are links to videos, articles, and NASA Kids ‘Club website. This section is intended to spark students’ interest and awareness of current as well as future opportunities.
Information on “How PLTW Addresses the Computer Science Job Challenge” is provided towards the bottom of the webpage. As one of the first elementary schools in southern California to implement PLTW Launch, we feel that this is an important feature to address in our webpage resource guide.
The last section of the webpage provides a link to Common Sense Media Education page titled: “K-12 Digital Citizen Curriculum”. This critical component to using technology as a digital citizen provides ratings, educational resources, and advocates for children, families, and schools.

Meeting the “T” in STEMM through Computer Science and Coding webpage resource guide is a living online document, which will be continually edited and updated throughout the school year by the authors. Current photographs, resources, articles, and events in regards to computer science, will continue to be maintained by the authors. Students access the webpage throughout the school day on their digital devices in particular to access their Studio.code.org classroom section online activities. Students, teachers, parents, and others have expressed their gratitude for this webpage resource guide throughout this project. Dominic Caguioa, Affiliated Trainer for Code.org’s Introduction to K-5 Computer Science Courses, tweeted his feelings of pride for the webpage resource guide and use of Code.org curriculum school-wide.
CHAPTER FIVE: DISCUSSION AND IMPLICATIONS

Current and future trends in STEM education are leading educators to promote and support the development and education of future computer scientists and knowledgeable 21st century learners as early as possible, beginning as early as kindergarten. Instruction in computer technology, especially learning to code at a young age will promote URM girls’ and boys’ positive attitudes towards STEM-related fields. Early exposure to coding will help students understand the value of computers and other digital devices as tools rather than simply play things. Since it can take as many as 25 years to create a computer-competent computer scientist, it is essential to start in elementary school. A robust computer science curriculum beginning in kindergarten will help ensure we fill the computer science gap of today and tomorrow.

The earlier we introduce young students to coding, the more comfortable and capable they will become in secondary school, when they are presented with more in-depth learning opportunities. Computer science education beginning in kindergarten and offered throughout the grade levels may be able to help increase the participation of underrepresented female and minority students in the field of computer science. Positive experiences in computer science at a young age are believed to motivate students to explore future opportunities in computer science. If we want to close the computer science gap and build a literate and functional population, we need to make computer science accessible and engaging for all students.

This Resource Guide is the first step to build awareness for all, educators, parents and most importantly students. Both public and private sectors acknowledge the need to provide all students with opportunities to develop an understanding of computer science and computational thinking skills. With the plethora of resources now available to all ages, introducing the
knowledge and skills of computer science can begin in early elementary school and throughout the grade levels. A growing number of statistics point to the need and demand for STEM related jobs, with computing towards the top. It has been made clear that the economic well-being of our country is becoming more and more dependent on a computing workforce. In order for our country to continue to be innovative and creative for the 21st century and beyond, a diverse workforce must have access to the knowledge of the “T” in STEM education.
REFERENCES


MERLOT Journal of Online Learning and Teaching Vol. 7, No. 4, December 2011 525 Digital Natives: Ten Years After Apostolos Koutropoulos University of Massachusetts Boston, MA 02125 USA.


Up to the Challenge. Retrieved on October 7, 2015 from,


ADDENDUM

JOINT PROJECTS ADDENDUM

Submitted by:
Sara Castro ID# 106749901
Maureen Straub ID#106724226

Project Title: MEETING THE ‘T’ IN STEMM THROUGH COMPUTER SCIENCE AND CODING

The Division of Responsibilities is as follows:

Sara Castro shall complete these specific and unique tasks:
1. Introduction/Research question (Individual)
2. Literature Review (Individual)
3. Joint Project methods
4. Joint project Resource Guide
5. Conclusion (Individual)

Maureen Straub shall complete these specific and unique tasks:
1. Introduction/Research question (Individual)
2. Literature Review (Individual)
3. Joint Project methods
4. Joint project Resource Guide
5. Conclusion (Individual)

Reviewed and Accepted by:

_________________________________
Dr. Susan Belgrad, Project Chair

_________________________________
Sara Castro

_________________________________
Maureen Straub

_________________________________
Associate Vice President of Research and Graduate Studies.