Spring 1-31-2020

Captains at the STEM of Their Own Ship: An Examination of Underrepresented Minority Student Participation in a Self-Directed, ICT After-School Intervention

Timothy Hicks
Georgia State University

ACCEPTANCE

This dissertation, CAPTAINS AT THE STEM OF THEIR OWN SHIP: AN EXAMINATION OF UNDERREPRESENTED MINORITY STUDENT PARTICIPATION IN A SELF-DIRECTED, ICT AFTER-SCHOOL INTERVENTION, by TIMOTHY ALEX HICKS, was prepared under the direction of the candidate's Dissertation Advisory Committee. It is accepted by the committee members in partial fulfillment of the requirements for the degree, Doctor of Philosophy, in the College of Education & Human Development, Georgia State University.

The Dissertation Advisory Committee and the student's Department Chairperson, as representatives of the faculty, certify that this dissertation has met all standards of excellence and scholarship as determined by the faculty.

____
Brendan Calandra, Ph.D.
Committee Chair

____
Jonathan Cohen, Ph.D.
Committee Member

____
Maggie Renken, Ph.D.
Committee Member

____
George Pullman, Ph.D.
Committee Member

____
Date

____
Brendan Calandra, Ph.D.
Chair, Department of Learning Sciences

____
Paul A. Alberto, Ph.D.
Dean, College of Education &
Human Development

AUTHOR'S STATEMENT

By presenting this dissertation as a partial fulfillment of the requirements for the advanced degree from Georgia State University, I agree that the library of Georgia State University shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to quote, to copy from, or to publish this dissertation may be granted by the professor under whose direction it was written, by the College of Education and Human Development's Director of Graduate Studies, or by me. Such quoting, copying, or publishing must be solely for scholarly purposes and will not involve potential financial gain. It is understood that any copying from or publication of this dissertation which involves potential financial gain will not be allowed without my written permission.

____

Timothy Alex Hicks

NOTICE TO BORROWERS

All dissertations deposited in the Georgia State University library must be used in accordance with the stipulations prescribed by the author in the preceding statement. The author of this dissertation is:

Timothy Alex Hicks
Department of Learning Sciences
College of Education and Human Development
Georgia State University

The director of this dissertation is:

Dr. Brendan Calandra
Department of Learning Sciences
College of Education and Human Development
Georgia State University
Atlanta, GA 30303

CURRICULUM VITAE

Timothy Alex Hicks

ADDRESS: 3605 Eagle Rise
Lithonia, GA 30038

EDUCATION:

Ph.D.	2019	Georgia State University Department of Learning Sciences Instructional Technology
M. A.	2004	Georgia State University Department of English Rhetoric and Composition
B. A.	2003	Georgia State University Department of English Rhetoric and Composition

PROFESSIONAL EXPERIENCE:

2012-Present	Graduate Research Assistant Georgia State University
2015-2018	Graduate Teaching Assistant Georgia State University
2014-2016	Website Administrator GATESOL in Action, Atlanta, GA
2013-2018	Website Administrator New Day LABS, Atlanta, GA
2004-2006	Graduate Teaching Assistant Iowa State University
2004-2004	Graduate Teaching Assistant and Graduate Assistant Georgia State University

PRESENTATIONS AND PUBLICATIONS:

Peer-Reviewed Journal Articles

Calandra, B., Renken, M., Cohen, J., Hicks, T. A., & Ketenci, T. (n.d.). An examination of a group middle school students' engagement and computational thinking outcomes during a series of afterschool computing activities. *Computer Science Education*. (Revised and Re-submitted)

Hicks, T. A., Cohen, J., & Calandra, B. (n.d.). Nine middle school students' participation in an inner-city, afterschool computing program: A case study with design implications. *Journal for Multicultural Education.* (Submitted)

Conference Presentations

Hicks, T. A., Ayer, T., Fuqua, A., & Cohen, J. D. (2016, Oct.). Urban Students' Motivation in a STEM-Related Afterschool Program. Presented at *Georgia Educational Research Association Conference*. 30.

PROFESSIONAL SOCIETIES AND ORGANIZATIONS

2016-Present	Association for Talent Development
2019-Present	American Educational Research Association

CAPTAINS AT THE STEM OF THEIR OWN SHIP: AN EXAMINATION OF UNDERREPRESENTED MINORITY STUDENT PARTICIPATION IN A SELF-DIRECTED, ICT AFTER-SCHOOL INTERVENTION

by

Timothy Alex Hicks

Under the Direction of Dr. Brendan Calandra

ABSTRACT

Recent studies have advocated for early adoption of Information and Communications Technology (ICT) in order to help a broader range of youth become creators rather than consumers of digital media, to open doors for opportunity in the lucrative technology sector, and to set them on a course for lifelong STEM/ICT learning. This study used data that was collected from a grant funded, multi-site, after-school program designed to help a group of students who are often underrepresented in ICT learn about computing through a unique instructional design for guiding students through the creation of mobile apps using a freely accessible block-based coding platform developed by MIT called App Inventor. The study employed a concurrent, triangulation mixed methods approach to data analysis. Data sources included participant-observer field notes, interviews, student artifacts, online surveys, and an assessment of outcomes *related to a construct called computational thinking. The purpose of the intervention and this proposed study was to examine whether participants in the program learned coding and related concepts, developed an interest in STEM/ICT subject matter, and gained an optimistic view of their abilities related to 21st century computing skills. In addition, the researcher hoped to identify which aspects of the instructional design may have facilitated progress towards these goals.*

INDEX WORDS: Computational Thinking, Mixed Methods, Scaffolding, Self-Directed Learning, Self-Efficacy, ICT

CAPTAINS AT THE STEM OF THEIR OWN SHIP: AN EXAMINATION OF UNDERREPRESENTED MINORITY STUDENT ENGAGEMENT, SELF-EFFICACY, INTEREST, AND PERSISTENCE IN A SELF-DIRECTED, ICT AFTER-SCHOOL INTERVENTION

by

Timothy Alex Hicks

A Dissertation

Presented in Partial Fulfillment of Requirements for the

Degree of

Doctor of Philosophy

in

Instructional Technology

in

the Department of Learning Sciences

in

the College of Education and Human Development

Georgia State University

Atlanta, GA
2019

DEDICATION

This dissertation is dedicated to my family, friends, colleagues, faculty, and creator.

ACKNOWLEDGMENTS

This dissertation is dedicated to my family for their ongoing support throughout my tenure as a doctoral student. A special thanks is extended to GSU friends and classmates who collaborated on the after-school project, helped answer various questions, took notes during my prospecuts defense, and provided moral support along the journey. Lastly, I would like to thank my dissertation committee, particularly my advisor, Dr. Calandra, for their vital feedback and help in navigating through this entire process.

Table of Contents

LIST OF TABLES .... **vii**

LIST OF FIGURES .... **ix**

1 THE PROBLEM **1**

- **The ICT Workforce in the United States** **2**
- **Career Interest and graduation rates in STEM/ICT** **6**
- **Starting Early With STEM/ICT** **8**

A Proposed Solution **13**

Purpose **15**

Assumptions and Limitations **16**

Terms and Definitions **17**

2 REVIEW OF THE LITERATURE **22**

Motivating Underrepresented Minorities in ICT/STEM **22**

- **Early Preparation** **24**
- **Community Building** **28**

The Benefit of After-School Programs **34**

Instructional Design **41**

- **Self-Directed Learning** **43**
- **Computational Thinking and Programming** **48**
- **Gamification of Learning** **54**
- **Culturally-Relevant Education** **55**

Culturally Responsive Computing **58**

Design Based Research **61**

Summary **67**

3 METHODOLOGY **68**

Rationale **69**

Context of the Study **71**

Setting ***71***

Participants ***72***

Data collection **74**

CT Quiz ***75***

Profile Survey ***77***

Interviews ***79***

Fieldnotes ***79***

App Projects ***80***

Data analysis **81**

Quantitative Data Analysis **83**

Correlation ***83***

Analysis of Variance (ANOVA) ***84***

Qualitative Data Analysis **88**

Summary **90**

4 RESULTS **91**

Outcomes Related to Computational Thinking 95

CT Quiz 95

Fieldnotes and Observation Data on CT... 99

Student Activity 99

Instructional Design 102

Summary of Participant Outcomes 105

Self-Efficacy Related to Participant's 21st Century Abilities.................................... 106

Profile Survey Data 106

ICT-21Q 107

ICT-21Q Correlation 108

Additional Survey Questions 108

Interview Data 112

Fieldnotes and Observation Data on Self-Efficacy.. 116

Affective Domain and Instructional Design... 116

Summary of Self-Efficacy... 120

ICT Affinity, Interest and Persistence 120

Profile Survey Data 121

SSS: Technology.. 122

SSS: STEM Career 123

Additional Survey Questions 124

Interview Data 129

Fieldnotes and Observation Data on Interest and Persistence 136

Affective Domain 137

Barriers and Conflict.. 139

Summary of Interest and Persistence in ICT 141

5 DISCUSSION 143

Quartile Compairison of Engagement and Outcomes Related to Computational Thinking.. 143

Participant Engagement.. 143

App Completion 144

CT Quiz 148

Quartile Comparison of Self-Efficacy, Affinity, Interest, and Persistence in ICT . 149

Self-Efficacy in ICT... 154

Affinity, Interest, and Persistence in ICT.. 156

Conclusions 157

Implications... 158

Suggestions for Further Research 165

Testing in different contexts.. 165

Collect more data on self-efficacy.. 166

Effect of file sharing and competiton 167

Changing data collection methods... 168

A longitudinal study on what students do next.. 169

REFERENCES **170**

APPENDICES **185**

Appendix A **185**

Appendix B **190**

Appendix C **192**

LIST OF TABLES

Table 1. Demographic Makeup of Scientists and Engineers in Related Positions in 2015. 3

Table 2. Demographic Makeup of Computer and Mathematical Positions in 2015 4

Table 3. Problem, Proposed Solution, Research Questions, and Data Sources 16

Table 4. Research Questions and Data Sources 69

Table 5. Student Tally and Self-Reported Student Demographics for the Main Rollout 73

Table 6. Concepts Being Studied and Their Respective Data Sources 75

Table 7. CT Quiz Questions and Their Corresponding CT Concepts and Cookbooks 76

Table 8. Research Questions and Data Sources 92

Table 9. Participants' Groupings into Quartiles Based on App Submissions 93

Table 10. Participants Who Completed 2+ CT quizzes 95

Table 11. Average CT Quiz Scores by Quartile 96

Table 12. CT Quiz Questions, CT Concepts, and the Relevant Cookbooks 97

Table 13. Participant Groupings Based on the Number of Apps Completed with Avg. Scores .. 97

Table 14. Number of Participants (n = 43) for the ICT-21Q and Avg. Scores for Quartiles 107

Table 15. Miscellaneous Data from the Pre-Test (Fall and Spring) Profile Survey 109

Table 16. Common Responses about the Participants' Motivations for Joining 110

Table 17. Common Responses about the Participants' Sentiments towards the Curriculum..... 110

Table 18. Percentage of the Responses as Positive, Negative, or Neutral Answers.................. 111

Table 19. Number of Interview Participants per Quartile... 113

Table 20. Common Interview Responses for Joining... 114

Table 21. Common Interview Responses for Things Learned 114

Table 22. Common Interview Responses for Participant Feelings.. 115

Table 23. Common Interview Responses for Making Real Apps 115

Table 24. Common Interview Responses for Peer Collaboration.. 116

Table 25. Number of Participants that Completed a Pre- and Post-Test for the SSS................ 122

Table 26. Averages for the Technology Portion of the STEM Semantic Survey....................... 123

Table 27. Averages for the STEM Career Portion of the STEM Semantic Survey................... 124

Table 28. Comparison of Responses Regarding Students' Feelings about Coding................... 124

Table 29. Additional Data from the Post-Test (Fall or Spring) Profile Survey......................... 126

Table 30. Common Short Answer Responses about Favorite Parts of the Intervention 126

Table 31. Common Short Answer Responses about Least Favorite Parts of the Intervention... 126

Table 32. Common Short Answer Responses about Improving Activities 127

Table 33. Number of Interview Participants per Quartile... 129

Table 34. Common Interview Responses about Interest in the Intervention............................. 130

Table 35. Common Interview Responses about Participant's Favorite Part of the Intervention 130

Table 36. Common Interview Responses about Participant's Least Favorite Part of the Interven-

tion 131

Table 37. Common Interview Responses about Attrition... 131

Table 38. Common Interview Responses about Persistence in Learning.................................. 132

Table 39. Common Interview Responses about Persistence in App Building 132

Table 40. Common Interview Responses about Improving the Intervention 133

Table 41. Common Interview Responses about Improving the Activities 133

Table 42. Student App and DIY Submission Numbers 144

Table 43. CT Quiz Data with Standard Quartiles and Quartiles Based on the Number of Relevant Apps Submitted.. 149

Table 44. Response Averages from CT Quiz, Surveys, and Interviews 1................................. 150

Table 45. Response Averages from CT Quiz, Surveys, and Interviews 2................................. 150

Table 46. Response Averages from CT Quiz, Surveys, and Interviews 3................................. 151

Table 47. Response Averages from CT Quiz, Surveys, and Interviews 4................................. 151

Table 48. Quartile Rankings Based on Average Responses 1 151

Table 49. Quartile Rankings Based on Average Responses 2 152

Table 50. Quartile Rankings Based on Average Responses 3 152

Table 51. Quartile Rankings Based on Average Responses 4 152

Table 52. Profile Survey Averages for Select Skills from the ICT-21Q 155

Table 53. Previous App Building and Programming Experience by Quartile........................... 160

LIST OF FIGURES

Figure 1. Chart of the demographic makeup of scientists and engineers in S&E and related positions in 2015 4

Figure 2. Percentage of the total workforce composed of computer and mathematical workers for select 5

Figure 3. Percentage of select ethnic workers in computer and mathematical occupations jobs relative to their ethnic group's workforce 5

Figure 4: Relative popularity of CS degrees for select ethnic groups from 2004-2014 7

Figure 5: Chart of computer science degrees earned for select ethnic groups from 2004-2014 7

Figure 6: Sample CT quiz question 77

Figure 7. ICT-21Q averages for each quartile 108

Figure 8. Percentage of students that completed each activity by quartile 145

Figure 9. Number of apps submitted by each quartile 146

1 THE PROBLEM

Pursuing an education and a career in information and communication technology (ICT) fields offers advantages due to the constant growth that science, technology, engineering, and math (STEM) industries have shown since the 1960s (National Science Board, 2018, p. 3-21), the possibility of a premium on earnings for workers with STEM degrees, the growing desirability of hiring STEM majors by employers in and out of a student's field of concentration (Noonan, 2017, pp. 2-4, 8), and the possibility for improvement of one's socioeconomic status (Intemann, 2009, pp. 250-251). Focusing exclusively on ICT, computer and math fields made up nearly half (49%) of all STEM employment in 2015 (Noonan, 2017, p. 3), arguably making study and pursuit of a career in computer science a desirable proposition for individuals seeking to open more doors for opportunities in the job market and build their socioeconomic status.

Social inclusion is a prime driving force behind research in ICT, which revolves around helping disadvantaged members of society be able to fully participate in the economy, education, health, recreation, culture, and every other aspect of society. ICT can help with social inclusion in issues of equity (e.g. technology and applications can assist individuals with disabilities) and access (e.g. technology can be used as a means for accessing information and education for people in more remote settings). Social participation being a main goal, ICT also has the ability to help improve cultural understandings through the exchange of ideas provided that people can use and access technology (Vrasidas, Zembylas, & Glass, 2009, pp. 11-14). One goal of ICT research is not just to help students learn to consume and analyze content but to participate in its creation as well (Barr & Stephenson, 2011; Gretter & Yadav, 2016).

In an information and technology driven society, diversifying the cohorts of ICT graduates that enter the workforce can help to influence decisions made regarding the design, development, and implementation of the types of technologies that are integral to our existence (Denson, Hailey, Stallworth, & Householder, 2015; Intemann, 2009; Noonan, 2017; Watson & Froyd, 2007). Unfortunately, to date, many minorities and women still remain underrepresented in the ICT workforce. While it is of utmost importance to bring more women into STEM and ICT fields and there are many high-quality initiatives in the US to close the gender gap in STEM/ICT education and employment opportunity, this section will focus on statistics and disparity in ICT and STEM between white and minority students and employees, regardless of gender. This choice was made by the author not because women were excluded from the aim of the study, but because the educational program and study within which it was imbedded was designed for minority girls and boys.

The ICT Workforce in the United States

According to the United States Census Bureau (n.d.-a; n.d.-b), the US population was estimated to be at 327,167,434 as of July 1, 2018. With the 2018 estimates, black/African American (13.4%), Hispanic/Latino (18.1%), Asian (5.8%), Hawaiian/Pacific Islander (0.2%), Native American/Alaskan (1.3%), and multi-racial residents (2.7%) made up 41.5% of the population while white Americans comprised the majority with at 60.7% of the population. In addition, there were over 17 million Science and Engineering degree recipients in the US workforce, and over 9 million workers actively employed in in science and engineering or related jobs in the United States, in 2015.

Based on population dynamics alone, it would seem difficult to be able to completely even out the number of minority and white graduates and employees in STEM related fields in the near future, however, building equal degrees of interest among each demographic is perhaps a more attainable and still worthwhile goal. For instance, relative to their makeup of the work-force, black/African Americans and Hispanic/Latinos had 4.71% and 4.16% of their scientists and engineers working in Science and Industry or related jobs compared to 8.43% of white Americans in the same position. On the other hand, computer and math related positions (STEM fields more closely related to ICT) had nearly equal representation for black and white workers relative to their entire workforce (2.14% and 2.53% respectively). See Tables 1 and 2 and Fig-ures 1-3 (Bureau of Labor Statistics n.d.-a, n.d.-b, and n.d.-c; National Science Foundation, 2017-d).

Table 1

Demographic Makeup of Scientists and Engineers in Related Positions in 2015

Ethnicity	Employed Workforce 16+ (BLS)	Employed Scientists and Engineers in S&E/Related Jobs (NSF)	Percentage of Ethnic Scientists and Engineers in S&E/Related Jobs Compared to the Total Workforce	Percentage of the Entire Ethnic Workforce that is a Scientist or Engineer in S&E/Related Jobs
Black	17,472,000	823,000	0.55%	4.71%
Hispanic/Latino	24,400,000	1,016,000	0.68%	4.16%
Asian	8,706,000	2,192,000	1.47%	25.18%
White	117,944,000	9,943,000	6.68%	8.43%
Native American/ Alaskan	NA	37,000	0.02%	NA
Hawaiian /Pacific Islander	NA	33,000	0.02%	NA
Multi-Ethnic	NA	231,000	0.16%	NA
ALL	148,834,000	14,274,000	9.59%	

Table 2

Demographic Makeup of Computer and Mathematical Positions in 2015

Ethnicity	Employed Workforce 16+ (BLS)	Employed in ICT or Related Careers (BLS)	Percentage of Ethnic ICT Workers in Total Workforce	Percentage of Ethnic Workforce in ICT
Black	17,472,000	374,000	0.25%	2.14%
Hispanic/Latino	24,400,000	NA	NA	NA
Asian	8,706,000	871,000	0.59%	10.00%
White	117,944,000	2,989,000	2.01%	2.53%
Native American/ Alaskan	NA	NA	NA	NA
Hawaiian/ Pacific Islander	NA	NA	NA	NA
Multi-Ethnic	NA	NA	NA	NA
ALL	148,834,000	4,369,000		2.94%

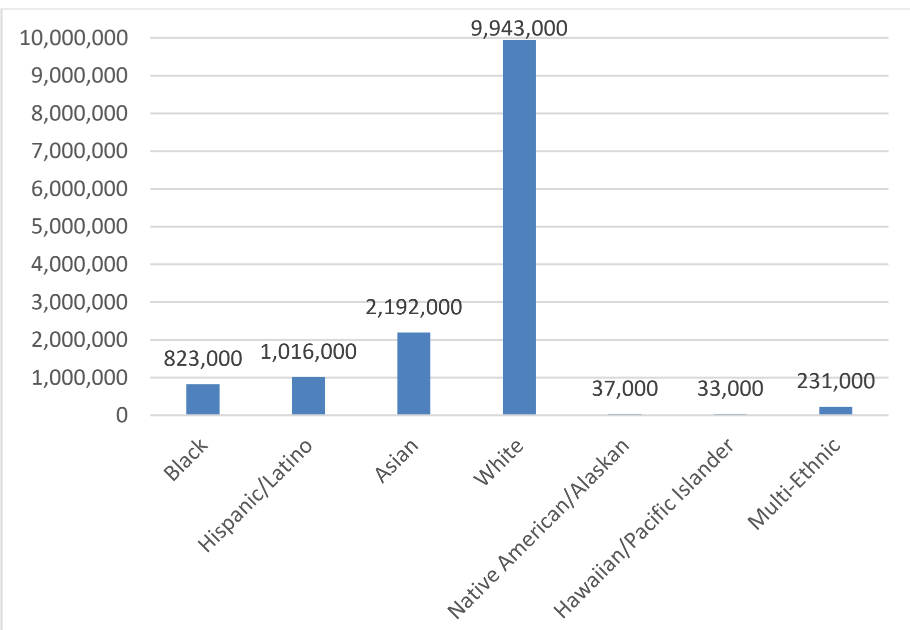

Figure 1. Chart of the demographic makeup of scientists and engineers in S&E and related positions in 2015

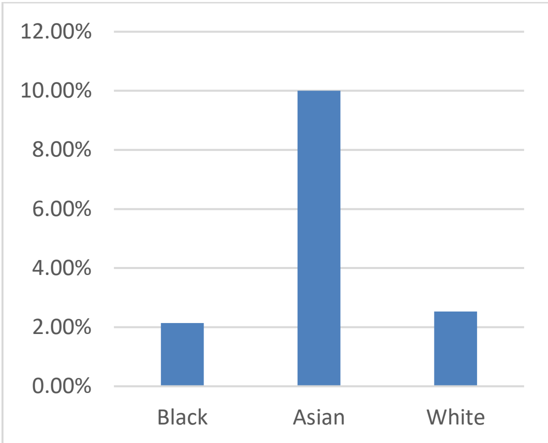

Figure 3. Percentage of select ethnic workers in computer and mathematical occupations jobs relative to their ethnic group's workforce

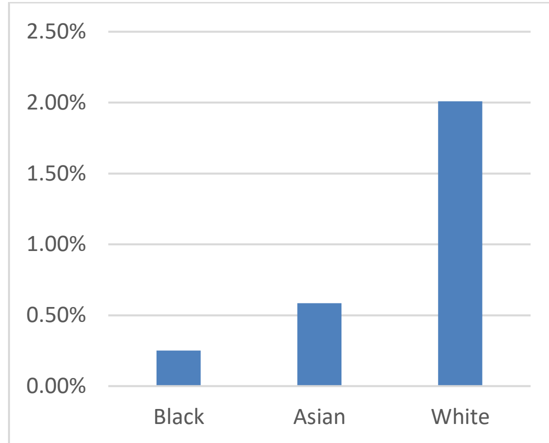

Figure 2. Percentage of the total workforce composed of computer and mathematical workers for select

The author feels it worth pointing out that ICT has already proven to have strong traction among many minorities traditionally underrepresented in STEM fields. The relative interest in ICT for black Americans is comparable to that of white Americans, with each group making up 2.14% and 2.53% of the workforce for their respective ethnic groups (Bureau of Labor Statistics n.d.-a, n.d.-b). In addition, according to Fayer, Lacey, and Watson (2017) of the U. S. Bureau of Labor Statistics, 7 of the 10 largest STEM careers are in computer and information systems, which include software/applications developers (largest), computer user support specialists, computer systems analysts, software/systems software developers, and network and computer systems administrators among others (p. 4). Among ethnic minorities, science and engineering job role preferences include information security analysts, computer support specialists, psychologists, industrial engineers, and computer systems analysts for black workers, while the most popular careers for Hispanics/Latinos were psychologists, economists, aerospace/aeronautical/astronautical engineers, and industrial engineers (National Science Board, 2018, p. 3-113).

Career Interest and graduation rates in STEM/ICT

Computer science shows somewhat more of an equalization in terms of the relative popularity among bachelor's degree seekers in each ethnic groups. For instance, in 2014, black, Hispanic/Latino, and Asian students all did comparably the same with 9.75%, 9.74%, and 9.56% of all degrees awarded respectively, but white students outpaced every other group with 55.6% of all bachelor's degrees. This difference is due to the size of each population. However, interest in computer science relative to the total bachelor's degrees earned by each ethnic group shows that black students outpaced white and Latino students, with 3.03% out of all bachelor's degrees earned by black students in the field in comparison to 2.70% of degrees among white, 2.58% among Hispanic/Latino, and 2.72% among Native American/Alaskan students. Ultimately, Asian students outpaced all others with 4.47% of their bachelor's degrees being in computer science. This shows that computer science is more or equally as popular among many underrepresented college/university students as it is for white students (National Science Foundation, 2017-b). Tables representing this data can be made available upon request.

Interest in ICT/computer science in academia also has comparable rates of interest among ethnic groups. Rates of attainment of bachelor's degrees in science and engineering fields in general show that interest amongst each ethnic group has mostly increased from 2004 to 2014. See Figure 4. On the other hand, computer science degree obtainment went down for most ethnic groups from 2004 but started to rebound in 2010 (National Science Foundation, 2017-b). See Figure 5.

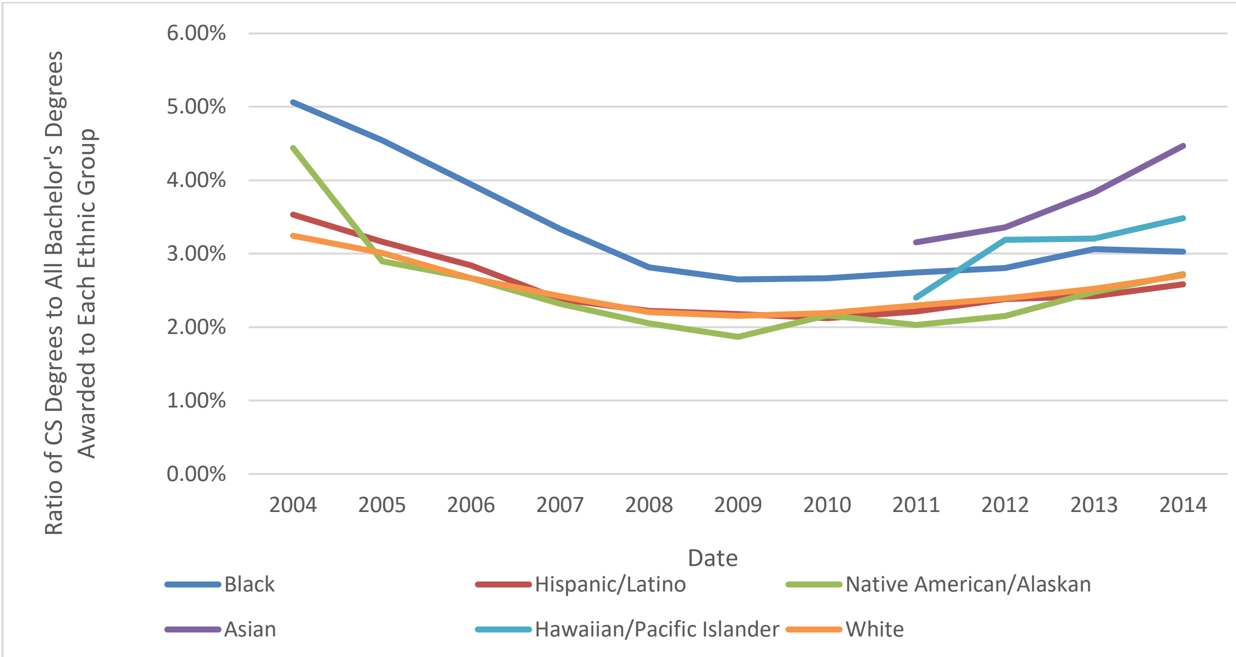

Figure 4: Relative popularity of CS degrees for select ethnic groups from 2004-2014

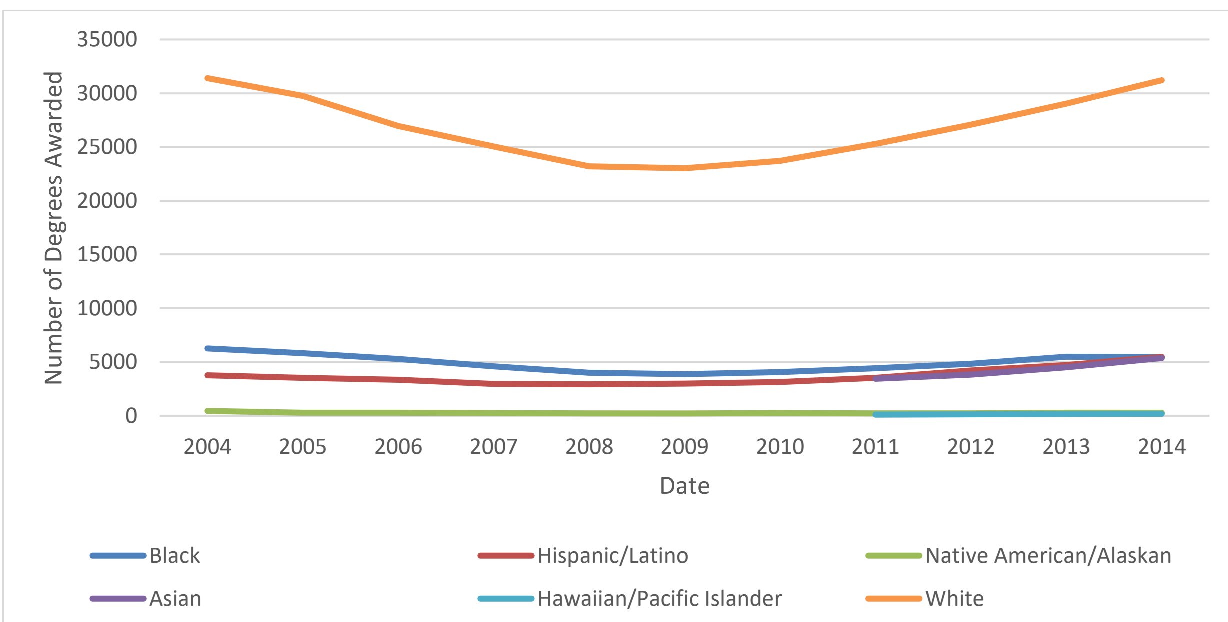

Figure 5: Chart of computer science degrees earned for select ethnic groups from 2004-2014

When considering student enrollment and graduation trends, it becomes obvious that sustaining interest is of great importance since attrition can be high among minority ICT/STEM students. For instance, in 2014, 40.4% of black freshmen entering a 4-year college/university planned to major in S&E in comparison to 45.1% of Hispanic/Latino, 54.2% of Asian, and 40.3% of white freshmen. More specifically, 4% of black freshmen are considering math, statistics, and computer science as their area of concentration in comparison to 4.1% of Hispanic/Latino, 8.7% of Asian, and 4.5% of white students. With exception to Asian students, who seem to be the most motivated to enter math and computer science fields, we can see mostly comparable levels of interest between the various ethnic groups coming into college.

However, it may be concerning that around 40% of black students plan on entering STEM fields but, going by trends from 2004-2014, only 29.39% on average end up getting a degree (National Science Foundation, 2017). This drop in interest is often due to students changing majors or dropping out of college. According to Chen and Soldner (2014), 48% of students entering STEM fields between 2003 and 2009 left the fields either by switching to a non-STEM major or dropping out of college/university entirely. Among the S&E fields, computer/information science had one of the highest attrition rates during this time period at 59%. However, students in ICT/STEM fields actually have a lower attrition rate than students outside of these fields, with non-ICT/STEM bachelor's degree programs seeing attrition rates at 56-62% (p. 14).

Starting Early With STEM/ICT

Curbing attrition begins by identifying the causes of attrition and providing adequate support to students that might be at risk of dropping out of ICT/STEM degree programs. Exploring ways that can be used to encourage minority students to become more invested in ICT/STEM fields thus has become a common theme for contemporary research (Andersen & Ward, 2014;

Campbell et al., 2014; Foltz, Gannon, & Kirschmann, 2014; McGee, 2015; Meador, 2018; Palmer, Maramba, & Daney, 2011; Price, 2010; Strayhorn, 2015; Zarrett & Malanchuk, 2005). Many researchers do this by testing instructional materials, learning practices, and various solutions with members of their target audience in order to come up with best practices for instruction, new theoretical frameworks for teaching ICT/STEM curriculum, or, more simply, to help students feel more interested in ICT/STEM subject matter and optimistic about their prospective pursuit of ICT/STEM education and employment (Andersen & Ward, 2014; Brown, Concannon, Mouza, Marzocchi, Pan, & Pollock, 2016; Burke & Dunn, 2003; Duran, Höft, Lawson, Medjahed, and Orady, 2014; Olszewski-Kubilius, Steenbergen-Hu, Thomson, & Rosen, 2016; Roberts et al., 2018; Sahin, 2013). This area of research is important for building interest in ICT/STEM fields among young, ethnic minority students, particularly at lower grade levels so that we capture their attention early, help them become invested, lifelong learners that want to persist in these content areas, and encourage them to start careers in these fields to help them improve their position in society, but also to benefit the larger community through their work and diverse perspectives. The importance of fostering interest in ICT among young, ethnic minority students revolves around a) giving them adequate preparation to meet the challenges they will face post-secondary school and b) helping the students establish an identity and a sense of belonging as a member of the ICT community.

On the issue of college preparation, Heck (2013) pointed out the feelings of Brown University students that responded to a survey on college level STEM coursework distributed by the student newspaper. Approximately 1/3 of the respondents reported feeling unprepared, of which minority students—45.3% of black and 61.2% of Hispanic in comparison to 30.2% of white students—felt the most affected. Similarly, Strayhorn (2015) administered a survey to black students in both historically black colleges and universities (HBCUs) and predominantly white institutions (PWIs), and he discovered that non-STEM majors had a stronger sense of self-efficacy regarding their ability to complete tasks. Only a small fraction of the participants in follow-up interviews (10 out of 38; 26.32%) felt adequately prepared after high school due to a lack of advanced/AP courses, less rigorous assignments, and inadequate resources (like computers) among other issues. The STEM students felt that self-efficacy was important if they were going to continue. Many other scholars have found that self-efficacy is equally as important as gaining STEM content knowledge, and it along with early exposure to STEM subject matter are important for persistence (Andersen and Ward, 2014; Hill, Corbett, and St. Rose, 2010; Wang, 2013; Zarrett and Malanchuk, 2005). For instance, Hill, Corbett, and St. Rose (2010) recommend building a healthy mindset towards learning and giving girls, in particular, assurance that they are capable of working on the subject matter, both of which are important for fending off any stereotype threats that they may face. Most importantly, they recommend helping students develop a "growth-mindset" by encouraging them to work hard and learn from mistakes when acquiring new information since it is the process of learning and building curiosity that is important at early stages more so than achieving perfection.

On the issue of gaining an identity and sense of belonging in an ICT/STEM field, many post-secondary students have identified the importance of encouragement and adult influence on their decision to persist in their field of study (Foltz, Gannon, & Kirschmann, 2014; Meador, 2018; Palmer, Maramba, & Daney, 2011). Foltz, Gannon, and Kirschmann (2014) interviewed minority graduate students and found that family influence (such as profession of parents), level of high school preparation, faculty support, experience outside of the classroom (clubs, internships, etc.), friendships and sense of community, financial aid, and personal drive were all factors that drove their success (pp. 3-8). Campbell, Skvirsky, Wortis, Thomas, Kawachi, and Hohmann (2014) similarly point out the desire of underrepresented students for instructors that have a genuine, invested interest in seeing them succeed and also help them early in their degree programs by mapping out pathways through the educational system that can lead to careers (p. 590).

Even before college and university, scholars like Koch, Lundh, & Harris (2019) point out the importance of family influence, specifically looking at the support they receive at home, in school, and in their after-school programs. The researchers found that parental support was seen to have a larger impact on girl's persistence in STEM and career ambitions than any other adult's input, and they recommend getting parents involved in STEM programs to provide, emotional, material, and role mode support to their children's STEM interests at home (pp. 265-266). Meador (2018) noted the influence of high school math teachers who were the catalyst in prompting some college students to major in math. Conversely, some students rebel against adult feedback, with one student describing how she never did well in science in high school but used the negative feedback from a school advisor that told her she would not amount to anything to try and prove everyone wrong by majoring in science in college (pp. 65-66). Positive reinforcement needs to be the theme of all efforts in encouraging persistence in ICT/STEM study. Strayhorn (2015) recommends faculty members make effort to get to know students and help them "'see' themselves in the curriculum" (pp. 59-60). Hill, Corbett, and St. Rose (2010) recommend exposing female students to positive, female role models in STEM fields and discussing the growing numbers of women that are working in STEM fields. They also state that schools should make a strong effort to recruit young women and take steps to welcome them into the community (with a positive environment, seminars, events, groups for women, etc.), even using Title IX laws to assure a fair environment with equal access for women (pp. 90-94).

Early recognition of student interest in ICT/STEM and preparation have been detailed by a range of researchers due to their importance in encouraging prolonged participation in ICT/STEM, particularly at the high school level (Foltz, Gannon, & Kirschmann, 2014; Meador, 2018; Strayhorn, 2015). While testing the means by which educators can build interest and guide ICT/STEM preparation in students, researchers have demonstrated positive outcomes from non-traditional activities outside of the classroom in STEM clubs, workshops, summer programs, assistantships, and many other communities of learning that value hands-on experience or tutoring (Andersen & Ward, 2014; Ericson & McKlin, 2012; Foltz, Gannon, & Kirschmann, 2014; Meador, 2018; Palmer, Maramba, & Daney, 2011; Price 2010; Sahin, 2013; Strayhorn, 2015). The success of these experiences comes from their ability to nurture a sense of community through peer collaboration and teacher/mentor encouragement, help students see future career options while pointing out the pathway to achieving that end, provide tutoring on specific subjects in which students are having difficulty, have students work on authentic tasks with subject matter experts, and raise awareness of how topics discussed in school have practical application (Campbell, Skvirsky, Wortis, Thomas, Kawachi, & Hohmann, 2014; Duran, Höft, Lawson, Medjahed, and Orady, 2014; Foltz, Gannon, & Kirschmann, 2014; Meador, 2018; Palmer, Maramba, & Daney, 2011; Roberts, Jackson, Mohr-Schroeder, Bush, Maiorca, Cavalcanti, Schroeder, Delaney, Putnam, & Cremeans, 2018; Wang, 2013). So there are informal ways through which students can gain entry into ICT/STEM discipleship beyond traditional classroom experiences, and these programs can help students get early preparation and boost confidence for ongoing study.

A Proposed Solution

The project within which this study was embedded sought to addresses some of the issues found in the literature related to building interest in ICT/STEM among young, underrepresented students by providing an entry level, after-school intervention for middle school students that introduces computational thinking (CT) and computer science while students create apps for Android smartphones by using App Inventor, a freely accessible, online app building platform created by MIT. App Inventor uses block-based coding to program the apps, giving the platform a "low floor/threshold" and a "high ceiling," meaning it is an easy way to get started with programming and create something quickly, but it also allows for the creation of more complex activities (Resnick, Maloney, Monroy-Hernández, Rusk, Eastmond, Brennan, Millner, Rosenbaum, Silver, & Kafai, 2009, p. 63; Repenning, Webb, & Ioannidou, 2010, p. 266). Inclusivity is the emphasis of App Inventor, for it "allows everyone – even children – to build fully functional apps for smartphones and tablets" (MIT App Inventor, n.d., para 1). App Inventor also allowed students to focus on designing an app and crafting the operations that were being carried out rather than writing and editing the more abstracted syntax of a programming language (Papadakis, Kalogiannakis, Orfanakis, & Zaranis, 2014, p. 5). Thus, participants could create projects rapidly, see how various elements of the code coordinated due to the visual cues of the programming language (such as with the shape and color of the blocks that automatically restrict illogical coding patterns), test their app at every stage of development directly on a phone or through an emulator, and finish with a product that was immediately usable for Android phones, giving real results that could be shared with others (Burnett, 1999; Rahman, 2018; Tsai, 2019).

This intervention strove to show students the real utility in what they were doing each session. The apps the participants developed were immediately usable on their phones upon completion, which students could then download and share with others. The end goal was for students to work through pre-designed activities in order to learn the App Inventor interface, including the function of various components and the coding syntax for adding interactivity. Then participants would be equipped to build personally relevant apps of their own design that appeal to their interests and could have a broader, positive impact on others.

The program within which this study was embedded met at nine different middle school sites during the 2017-2018 school year. The intervention predominantly served middle school students that identified as black/African American, Hispanic/Latino, or mixed ancestry in a major, metropolitan city in the southeastern United States. It allowed students to meet with other members of their cohort. These cohorts typically consisted of 6-8th grade students, although high school students could opt to join at sites that served grades 9-12. While the students were thought to be capable of working through the intervention's curriculum on their own, facilitators and mentors were on hand to provide adequate support and scaffolding during the time that participants became accustomed to the content and available resources. Sessions were facilitated by 6 graduate student researchers (5 black males and females and 1 white female) serving as participant-observers, several black/African American undergraduate students majoring in computer science or related fields from 3 historically black colleges and universities (HBCUs), and teachers who were participating in the larger after-school program. Intentionally including ICT/STEM facilitators/mentors who culturally identified as most of the participants did was one design aspect meant to alleviate stereotype threats and implicit bias experienced by some students in STEM fields (Meador, 2018).

While there is a large and growing body of work on STEM-based after-school activities and even app building through the use of block based coding for young students (Brennan & Resnick, 2012; Ericson & McKlin, 2012; Maloney, Peppler, Kafai, Resnick, & Rusk, 2008; Sáez-López, Román-González, & Vázquez-Cano, 2016), this after-school computing intervention differed in its a) focus on the App Inventor platform, b) progression from guided to more open, less scaffolded activities, c) emphasis on student-centered, self-directed learning strategies rather than traditional, teacher-centered formats, and d) collaboration between participants and mentors from outside of the participating schools. This instructional design together with an emphasis on using a block-based programming language were unique aspects of the intervention that the researcher found worthy of investigation.

Purpose

The purpose of this study was to determine how aspects of the aforementioned instructional design may or may not have influenced participants' knowledge and attitudes related to computer science. In keeping with the project goals and objectives, the researcher's expectation was that participants who completed more apps should have more knowledge of computational thinking concepts, higher self-efficacy related to 21st century abilities, more positive opinions of ICT, and a greater motivation to persist with ICT/STEM than those who participated to a lesser extent. The study was guided by the following questions:

1. Was there a relationship between participant engagement (i.e., the number of apps completed) and outcomes related to Computational Thinking?
2. Was there a relationship between participant engagement and their reported belief in their 21st Century abilities?

3. Was there a relationship between participant engagement and their opinions about ICT subject matter and/or their desire to persist in ICT?

To answer these research questions, this study used a concurrent, triangulation mixed methods approach to data collection and analysis (Hanson, Creswell, Clark, Petska, & Creswell, 2005). Data sources included participant surveys, a multiple-choice quiz, student and teacher interviews, researcher field notes, and student artifacts. This broad selection of data sources was aggregated in order to provide a broad overview of participants' daily activities, highlight any change in self-reported interests and motivations, and provide any insight into participants' ability to transfer what they learned from the guided, instructional packets into other contexts, including their own unique projects. See Table 3 for a matrix that maps the problem and proposed solutions to the research questions, and data sources in the current study.

Table 3

Problem, Proposed Solution, Research Questions, and Data Sources

Problem	**Proposed Solution**	**RQs**	**Data Sources**
Lack of engagement and preparation in ICT at a young age.	ICT/CS content knowledge acquisition through app building	Q1	• Student artifacts • Quiz results • Fieldnotes
Low self-efficacy in ICT related work.	Scaffolding Self-directed learning Peer collaboration Low floor/high ceiling design	Q2	• Survey results • Interviews • Field notes and reflections
Low interest in ICT/STEM subject matter and low persistence through academia and into an ICT/STEM related career.	Engagement in culturally relevant ICT activities Mentor support Low floor/high ceiling design Peer collaboration	Q3	• Survey results • Interviews • Field notes and reflections • Student artifacts

Assumptions and Limitations

The underlying assumption of this study is that participants who engaged in more hands-on app building activities would have a greater understanding of computational thinking, a more positive opinion towards ICT/STEM subject matter, a stronger sense of ICT-related self-efficacy, and a greater desire to continue participating in similar activities. This was examined by grouping participants into quartiles based on the number apps that they completed and comparing results across quartiles. The researcher also hoped to uncover which aspects of the instructional design may or may not have supported the process of gaining knowledge, improving self-efficacy, and/or increasing interest in ICT. The major limitation of the current study was the number of students who provided complete data sets. This was due to the intentionally voluntary, informal, and flexible nature of the intervention. For example, many students joined the program late or left the program early and did not always participate in all pre-tests, post-tests, and interviews simply because they were not required to do so. While the results of this study are not necessarily generalizable, it is hoped that conclusions reached in this dissertation can be used to guide future iterations of the after-school computing intervention under examination. In addition, with the study reported here, and future work informed by this study, the researcher intends to add to the literature on instructional design for STEM/ICT interventions aimed at fostering sustainable and lifelong interest in STEM/ICT among underrepresented minorities in the United States.

Terms and Definitions

Key terms and concepts used in this study include the following:

Computational Thinking (CT) – Computational thinking is the process of learning to think like a computer scientist by understanding the skills and concepts that are fundamental to computer science (Gretter & Yadav, 2016, p. 511). Ultimately, it is a way of using computers to build tools that can then be used to solve problems in the world; the tools are created using concepts that guide their production (Barr & Stepheson, 2011, p. 51). The process of creating artifacts is the primary concern of computational thinking, which consists of concepts and practices that computer scientists employ when developing apps or other projects. Computational thinking concepts are foundational ways for mapping code in many programming languages. Computational thinking practices are the activities with which computer scientists engage when developing programs, such as working incrementally and testing/debugging while they work (Brennan & Resnick, 2012, pp. 3-7).

Four CT concepts are emphasized in this study's testing materials and in many of our more advanced activity packets:

- *Events* — Events are processes that occur after a component triggers something in the program to happen (e.g. pressing a button may cause music to play). The reaction to an event may be a chain of actions that happen in sequence—with one thing occurring after another—or in parallel—sequences that occur simultaneously (Brennan & Resnick, 2012, pp. 3-4).
- *Variables* — In programming, a variable stores a value that is used during a computation (Kernighan & Ritchie, 1988, p. 6). For example, a game app might display a score in a text box and update the score whenever the user completes different tasks. A variable labeled as "score" in this instance can be created and then called in the code multiple times to do things like tally up points whenever a specific action occurs, print the updated score into the text field, and reset when a game is restarted. The variable stores the score of the game and is called multiple times in the coding to be edited based on how the game is played.

- *Loops* — Loops are a programming expression that cause a sequence of code to repeat multiple times or endlessly until a condition is met or a programmed limit is reached (Brennan & Resnick, 2012, pp. 3-4). For instance, an app might cause an MP3 file to play endlessly until a user navigates to a different screen).
- *Conditionals* — Conditional programming enhances interactivity by basing options around whether or not various circumstances are valid/have come into some alignment (Brennan & Resnick, 2012, p. 5). For example, a quiz app can make a warning message appear if a user chooses an incorrect answer or it can proceed to another question if the user chooses the correct answer).

Information and Communication Technology (ICT) — Information and Communication Technology (ICT) revolves around the use of technology for gathering and communicating information. ICT incorporates science and technology by looking predominantly at digital modes of communication (like cell phones, video, wireless networks, etc.) and ways computer and information scientists solve problems and develop solutions (Moursund, 2005, pp. 4, 6). Similarly, when reporting on the ICT sector, the National Center for Science and Engineering Statistics puts emphasis on ICT manufacturing (computer and electronics products, software publishing, telecommunications, data hosting, computer systems design, and related technological fields that are not traditional paper publishers (Shackelford & Jankowsi, 2016, p. 1). The emphasis of ICT leans towards the development and usage of artifacts to not just communicate information but also to solve problems. Encouraging students to become content creators for mobile phones is the major emphasis of this intervention.

Mixed Methods Research — Mixed methods research is a method of inquiry that combines quantitative and qualitative methods of data collection, which allows researchers to gather data with more precision through the use of questions with narrow, pre-determined responses through quantitative analysis (such as surveys with multiple choice questions) as well as broader, open-ended data through qualitative methods (such as interviews). Mixed data sets can be used together by attempting to quantify the qualitative data (or vice versa) or by using data from one method to guide later data collection efforts through the other method (Creswell & Creswell, 2005, pp. 317-318).

Scaffolding — Scaffolding is an approach to learning where the teacher gradually gives students more responsibility for their own learning. Rather than give explicit instructions every time a student needs help, a teacher will serve as more of a coach by modeling behaviors that students can exhibit in order to learn on their own, helping to break down complex tasks into manageable discrete units, and simply prompting students to work more independently. The goal is to help students learn strategies for solving problems while they also gain content knowledge on a specific topic (Blumenfeld et al., 1991, p. 371).

Self-Directed Learning — The goal of self-directed learning (SDL) is to put students in charge of the planning, process, and evaluation of learning while instructors serve as facilitators that help students with tasks, such as making decisions, finding resources, and doing a needs assessment (Merriam, Caffarella, & Baugartner, 2007, pp. 110, 113). Going further, SDL can also be used to for emancipatory learning and social action by having students evaluate the historical and cultural significance of their activities and allowing them to make their own decisions about how to apply what they are learning and giving them easy access to resources to help them expand their knowledge (p. 107-109).

Self-Efficacy — Self-efficacy is the belief in one's ability to be successful in a subject that is being studied (Andersen & Ward, 2014, p. 218). According to Bandura (1995), "perceived self-efficacy refers to beliefs in one's capabilities to organize and execute the courses of action required to manage prospective situations" (Bandura, 1995, p. 2). Most simply stated, self-efficacy, then, is one's belief in his or her ability to accomplish tasks required for a given situation. Many scholars have explored the role of self-efficacy as a predictor for determining if students will persist and be successful in a field of study (like Andersen & Ward, 2014; Strayhorn, 2015). In a comparison of data between different 9th grade ethnic groups studying STEM subjects, Andersen and Ward (2014) found that self-efficacy, attainment value, and intrinsic value were predictors for white students along with attainment value and achievement while achievement, attainment value, and intrinsic value served as predictors for black students (p. 225). This could signify that black students value their grades and their personal interest in learning a subject matter and use that as motivation to persist. However, other studies, such as Strayhorn (2015), point out that having a sense of preparedness and confidence in one's ability to succeed plays an important role in getting students to achieve their long-term goals and not giving up in college (p. 54). MacPhee, Farro, and Canetto (2013) found that students with STEM minority statuses (ethnicity, gender, SES, etc.) had lower perception of their ability even if their performance was similar while students with multiple STEM minority statuses performed lower on their research measures. However, students with multiple STEM minority statuses benefitted more from participation in a mentoring program in at least two areas: critical thinking and perceived creativity (pp. 362-363).

2 REVIEW OF THE LITERATURE

The literature review provides context for the current study by discovering how previous scholarship has influenced our approach in designing and theoretically framing this after-school computing research project as well as our data collection and analysis process. This investigation fits into a larger, concerted effort in building interest in ICT/STEM subject matter—specifically computer science and app building—among young, minority students. First, this chapter will compile information pertaining to the factors contributing to entry and persistence in studying STEM curriculum before discussing some of the learning theories that went into the development of the curriculum behind this research project: self-directed learning, computational thinking, gamification of learning, and culturally relevant education/culturally responsive computing. Second, the methodology for collecting data will be discussed in more detail as this study is a more concise version of design-based research that uses a mixed methods approach to data collection and analysis in order to guide refinement and further testing of the after-school computing program's intervention while also using principles of grounded theory to identify and develop any emerging learning theories that emerge from our data.

The following 10 areas are explored: Motivating Underrepresented Minorities in ICT/STEM, the Benefit of After-School Programs, Self-Directed Learning, Computational Thinking and Programming, Gamification of Learning, Culturally Relevant Education and Culturally Responsive Computing, Design-Based Research, and Grounded Theory.

Motivating Underrepresented Minorities in ICT/STEM

The main purpose of this study's after-school program is to build interest in programming through app building at an early age, thereby developing what we hope becomes a long-term interest in ICT/STEM subject-matter that carries over into post-secondary education. Building equity in ICT/STEM fields begins by creating a more favorable attitude towards ICT/STEM and equipping students with content knowledge so that they will be more likely to enroll into STEM degree programs and perhaps even persist to a career. The benefit of this after-school intervention comes from 1) the immediacy of creating apps that can be used on Android phones and 2) the gain of transferrable skills—learning and using computational thinking (CT) concepts as they build apps—that can be used as a starting point towards more traditional programming languages like Java. This study will look at the instructional design of the after-school intervention and its effectiveness in building interest in ICT/STEM, encouraging persistence, improving student's perception of their ability to work with 21st Century Skills, and teaching CT concepts. The importance of this area of research comes from the lack of college preparation that many underrepresented minority (URM) students feel upon entering post-secondary STEM programs and the importance of revealing other barriers that many perceive to block their entry or lead them to attrition.

Many scholars have sought to uncover the causes for attrition and strategies to encourage persistence among minority students in STEM fields, laying the groundwork that can lead to the student's continued participation and improved feeling of self-efficacy in their field (Andersen & Ward, 2014; Campbell et al., 2014; Foltz, Gannon, & Kirschmann, 2014; McGee, 2015; Meador, 2018; Palmer, Maramba, & Daney, 2011; Price, 2010; Strayhorn, 2015; Watson & Froyd, 2007; Zarrett & Malanchuk, 2005). Most researchers tend to hear directly from current students or recent college graduates in order to gain feedback on their experiences in K-12 and post-secondary STEM education through observations, surveys, and interviews while other researchers may indirectly look for patterns of student behavior while reviewing quantitative data from longitudinal studies, school enrollment records, information on declared majors, standardized test scores, and/or some other independently gathered but relevant sources of information. Most studies on minority students in STEM tend to most commonly point to three overarching areas that affect persistence into college and beyond: **early preparation** (opportunities to learn in and out of the K-12 classroom in order to cultivate interest), **sense of belonging** (self-efficacy, stereotype threat avoidance, peer support and respect), and **adequate guidance** (invested faculty support, role models, early information about careers, clear pathways towards completing education and career goals, etc.). All three of these concepts can be put in place early in the lives of URM students to prep them for continued study.

Early Preparation

On the issue of inadequate preparation, Heck (2013) pointed out the feelings of Brown University students that responded to a survey on college level STEM coursework distributed by the student newspaper. Approximately 1/3 of the respondents reported feeling unprepared, of which minority students—45.3% of black and 61.2% of Hispanic in comparison to and 30.2% of white students—felt the most affected. Getting students prepared to study STEM in college must begin much earlier while students are young so that they are not spending so much of their time trying to catch up with their peers rather than already being in pace with them, and its importance, particularly for high school students, has been detailed by a range of researchers (Foltz, Gannon, & Kirschmann 2014; Meador 2018; Strayhorn 2015).

Strayhorn (2015) administered a survey and conducted interviews with black students in various universities, with one third of the students being STEM majors, and he found that STEM majors had a weaker sense of self-efficacy than non-STEM majors (though they felt it was important) and they mostly did not feel adequately prepared for study in their major due inadequate preparation (lack of advanced/AP courses, less rigorous assignments, insufficient resources, etc.). Researchers like Wang (2013); Andersen and Ward (2014); Zarrett and Malanchuk (2005); and Hill, Corbett, and St. Rose (2010) also identified early immersion in STEM and strong self-efficacy as important for persistence or factors that could serve as predictors for one's future in STEM.

Using a 4-year longitudinal study (following a cohort from the 10th grade to 2 years into college/university), Wang (2013) noted the importance of targeting URM students at an early age in order to build their sense of achievement in math and their self-efficacy, which are two variables that were found to impact their desire to pursue a STEM degree after high school. Math achievement established in the 10th grade (with 10th grade data used as a proxy for earlier influence in general) had an impact on 12th grade math self-efficacy in particular, while early math attitudes (such as self-efficacy) influenced STEM pursuit generally. The data did not specifically reference computer science, but the study results favor early exposure of science and math for URM students. Wang also recommends working to improve self-efficacy among female students (pp. 1106-1108). However, Wang is careful to point out that "STEM participation cannot simply be resolved by offering more math and science to underrepresented minority students" in high school (p. 1110). The onus then becomes one of finding practices that work to heighten the student's sense of achievement and belief in their ability to succeed at an earlier age, which is echoed by other scholars.

Andersen and Ward (2014) sought to identify predictors for persistence in STEM fields, and they found that math achievement, science intrinsic value (seeing worth in learning science for its own sake), and science attainment value (believing science fits in with their identity and long term goals) as significant predictors of one's plans to persist among black students. Similarly, Hispanic students have STEM utility value (seeing the importance for one's future academic or career plans) and science attainment value as predictors (p. 225). For black students, science intrinsic value was a significant predictor though the mean score was very low (-.15) compared to Hispanic (.32) and white (.40) students, which could indicate that black students generally do not find school science classes to be personally relevant or useful for college or career (p. 235). Attainment value was found to be a predictor because it links a student's identity with his or her perception of science as a domain. Some students are willing to conform their identity to the expected identity of a member of a STEM community (p. 236). Andersen and Ward recommend using methods to improve student's identities as scientists and improve curricula to show qualities that students tend to value (p. 237). The authors also recommend helping students a) find "congruence between STEM identities and students' identities," b) become more aware of the utility of math and science courses for achieving their future goals, and c) feel more interested in these subjects (p. 237). Though Andersen did not find self-efficacy to be a predictor of future achievement for URM students, the importance of seeing worth in what one is learning and making some achievements in a given ICT/STEM field is important for students at an early age.

Zarrett and Malanchuk (2005) similarly were interested in exploring which social psychological factors influence students decision to persist in studying information technology (IT) and pursue a career. Focusing on differences between men and women and black and white students (since other ethnic groups had low participation in their study), the researchers looked at the student's plans for a career in soft (help desk, tech journalism, etc.) or hard IT (programming, etc.) by using data from the Maryland Adolescent Development in Context Study (MADICS), a longitudinal study that collected data from students 6 times (from the 7th grade to 3 years after grading high school) between 1994 and 2000 (pp. 66-69, 71). The majority of students gave no consideration to a career in IT, but males were more likely to consider an IT career than women. Interestingly, black males were more likely to consider a career in IT than white males, but white males were more likely to consider an IT major in school, which is increases the likelihood of a career in IT. Furthermore, there was little difference between ethnic groups and genders in pursuing a soft IT career, but men favored hard IT careers by a larger margin (pp. 69-70). This study illustrates the importance of discussing career opportunities in STEM fields with URM students and making it clear how a degree program can lead towards one's career goal.

Zarrett and Malanchuk (2005) note that cultural discrimination more so than overt discrimination may be keeping women out of the IT field if they just consider it something that is for guys or they just perceive their ability in coursework is lower even if it is not the case. For this reason, they recommend intervening early to nurture interest in ICT/STEM fields to develop their self-concepts of their ability and remove barriers to their entry in that regard (pp. 77-78, 80). In their study, black males had a positive attitude towards computers and the IT field, but concerns about discrimination tend to be a barrier for pursuit in various fields if they perceive they would be one of very few black males in the industry. This points out the need to build a welcoming community for students considering a career in ICT/STEM fields, with the researchers noting that providing sufficient support and guidance along with enrollment in IT courses could help black women pursue an IT career (p. 79). This study does not overtly call for more research on the role of mentors or role models in pursuing a career in ICT/STEM, but such influences may have an impact on students if they provide continuous support.

Hill, Corbett, and St. Rose (2010) similarly focus on the importance of fending off any stereotype threats that women may face, particularly in the stigma that they are not capable of working in ICT/STEM subject. The authors recommend presenting the real-life application of what is taught in STEM courses and encouraging girls to take advanced math and science classes when available while making performance standards and expectations clear to reduce uncertainty about what a class's performance metrics indicate. The emphasis, though, must be on changing one's mindset to a "growth-mindset" where one learns from mistakes and focuses more on growth than on perfection. They also argue for addressing the issue of stereotype threats directly and assuring girls that there is a lack of difference in performance between the sexes but women just need to develop a growth mindset to succeed in the field (pp. 90-94). This advice is applicable to boys as well and can be used in mixed classes, but it is especially important for helping girls persist.

A lot of the research on building on early preparation in and out of the classroom discusses the effect on persistence. A lot of information about persistence comes down to the students' attitudes and perceptions towards ICT/STEM subject matter, with the threat of stereotypes—often stemming from a lack of a sense of belonging in ICT/STEM communities—being a major barrier. Therefore, community building, invested mentorship, and teacher/adult support is important areas of research as well.

Community Building

Existing literature on helping URM students have purported that maintaining student interest in ICT/STEM can be supported by providing adequate coaching, building a welcoming community of scholars, and helping students feel they belong in that community. College students particularly want to socialize and be involved with campus life beyond taking classes and students can support each other by studying together as well (Foltz, Gannon, & Kirschmann, 2014, p. 7), while coaching can come from a combination of instructors, role models, or parents that act as a support system. Developing a community and providing students support should happen early (even before college), and some of the requirements should include extending basic curtesy to URM students so they do not feel alienated, especially since stereotype threat can become a reality if other students close them off rather than welcome them. Stereotype threat is "the self-perceived concept and the apprehension caused by the idea that engaging in certain behaviors may confirm negative attributes commonly associated with minority group membership" (Meador, 2018, p. 63). Stereotype threat can surface when students or faculty assume URM students are unprepared when that is not the case or when students or faculty refuses or reluctantly gives support to an URM that needs help in some area.

For instance, Strayhorn (2015) surveyed and interviewed students who identified the importance of feeling a sense of belonging in STEM fields, with some reporting that they did not feel valued when people did not learn their name (a basic courtesy), confused them with one of the few other minority students in a program, or assumed they did not know course material (pp. 46, 50-60). Similarly, minority students have expressed unhappiness when white peers do not want to work with them in class or include them in study groups (Heck, 2015). Female students have expressed experiencing pushback from male students that might tell them to switch majors or similarly assume that they are not as smart as the other male students (Meador, 2018, p. 67). Minority students may come into a degree program with optimism, but some have stated that they were informed that they were at a disadvantage, which they were not aware of beforehand, and that negatively affected their mentality going forward, making it difficult for them to perform or take exams (Heck, 2015).

However, many URM students have discussed the positive impact of teachers in K-12 and postsecondary school that gave them encouragement, checked on them, and earnestly wanted to help them out (Palmer, Maramba, & Daney, 2011; Foltz, Gannon, & Kirschmann, 2014; Meador, 2018). A student's community has a lot of sway in whether URM students feel encouraged or discouraged in their academic progress, and much of it has to do with just the level of dignity that students are given on campus. The challenges of interpersonal relationships can be mitigated by providing strong support and guidance from faculty, which can motivate students to persist and focus their energy on building expertise.

Unfortunately, this research project was only meant to provide a curriculum on app building for a one-year interval at various schools and not provide ongoing mentorship in information communication technology or computer science. Nevertheless, the importance of providing a warm environment is fundamental as is our role of assuring students that they are capable of completing the activities. Both of these functions are a bases for building a community of scholars and nurturing lifelong learning.

Educational psychologists have already shown the relevance of community in terms of building intrinsic motivation through Maslow's 5 hierarchy of needs, which include having physiological needs met, providing a safe environment, encouraging a feeling of belonging, building self-esteem and gaining esteem from others, and nurturing the self-actualization (the drive to become all that one can be) of students. When students are not exerting energy and concentration on the lower level needs, like belonging and esteem, then they have more time to focus upon self-actualization (Ormrod, 2008, pp. 458-460), which could be viewed as ongoing, lifelong learning. Simple encouragement and respect from instructors is a minimal need that all students should have met by faculty (and hopefully by peers), and the effort can be reciprocated with students noticing the effort and attributing faculty care as a spur to try harder and not let them down (Foltz, Gannon, & Kirschmann, 2014, pp. 5-6). Some URM students encourage instructors to go further by having a genuine, invested interest in seeing them succeed and also by pointing out pathways through the educational system that can lead to careers early in their degree programs (Campbell, Skvirsky, Wortis, Thomas, Kawachi, & Hohmann, 2014, p. 590). Seeing the bigger picture—the career options that can result from following a path of study—can serve as an extrinsic motivator and keep students focused. To further aid students, colleges/universities can also help by making students more aware of financial aid, scholarships, and extra-curricular STEM activities that are available (Foltz, Gannon, & Kirschmann, 2014, pp. 6, 8). While this research project cannot take that step and help students financially, we can help participants see the bigger picture of a career in ICT/STEM and we do so, in part, by providing videos from black/African American workers in ICT/STEM fields describing their careers and their path to that end.

The issue of community building begins in K-12 and at home. Koch, Lundh, & Harris (2019) point out the importance of family influence in a study where they looked at various socio-cultural factors that can influence STEM persistence among underrepresented minority girls, specifically looking at the support they receive at home, in school, and in their STEM vs non-STEM after-school programs (pp. 244, 246, 249). Using interview feedback from the students, parents, teachers, mentors, and after-school staff, the researchers created a profile on each student (six girls in total) based on their levels of support (emotional, material, and social), personal interests, enjoyment, self-efficacy, mindset (growth mindset, which assumes effort needed to learn and improve, vs natural ability, which assumes that a student is naturally talented or not and may lead to a less disciplined approach to learning), and career ambitions (pp. 246, 250-251). Ultimately, parental support was seen to have a larger impact on the girl's persistence in STEM and career ambitions than any other adult's input, with children particularly influenced by the careers of their parents and close relatives (pp. 250-265). The researchers recommend getting parents involved in STEM programs to provide, emotional, material, and role model support for a child's STEM interests at home (pp. 265-266).

Meador (2018) also goes in-depth when discussing the impact of family influence on persistence, but noting that teachers play a strong role as well. For instance, three survey respondents and one interview participant pointed out high school math teachers as influences for their decision to major in math. Conversely, one student mentioned never doing well in science in high school and used the negative motivation of an advisor telling her she would not amount to anything to try and prove everyone wrong (pp. 65-66), but this kind of negative motivation is not necessarily commonplace. Having family members with degrees in STEM influenced some students while others cited faculty members as role models or as providing critical support (pp. 66-67).

Foltz, Gannon, and Kirschmann (2014) interviewed 8 minority graduate students, purposefully sampled, and some faculty and staff in order to better understand what factors helped the students persist in STEM coursework. Family influence, high school preparation, faculty support, experience outside of the classroom, friendships and a sense of community, financial aid, and personal drive were all factors that drove their success (pp. 3-8). Two of the students identified their family as an influence to study in STEM fields due to a parent or close relative working as science teachers/professors. The majority of the students mentioned having support from their family to succeed though at least one mentioned that family negativity spurred her to prove them wrong (pp. 4-5). Faculty support is also important, and students noted that faculty that cared about them and got to know them through regular meetings could make a big difference and spur students who do not want to let them down (pp. 5-6).

Hill, Corbett, and St. Rose (2010) discuss the importance of community building for female students, recommending we expose students to female role models in STEM fields and discuss the growing numbers of women that are working in STEM fields. They also state that schools should make a strong effort to recruit women and take steps to welcome women into the community (positive environment, seminars/events/groups for women, etc.), even using Title IX laws to assure a fair environment with equal access for women (pp. 90-94).

While researchers have pointed out that many minority students tend to band together in STEM fields when there are few other minority students attending their classes and they often attend study groups together (Meador, 2018; Palmer, Maramba, & Daney, 2011), this should not serve as a substitute for faculty and other students fulfilling their role for creating a welcoming environment if schools have any desire to curb attrition rates. Strayhorn (2015) issues an important warning by explaining that students that continue to feel socially isolated or unwelcomed were more likely to change majors or consider dropping out of school. Strayhorn recommends faculty members make effort to get to know them and help “students ‘see’ themselves in the curriculum” (pp. 59-60).

It is also worth noting that though it is not always possible given the personnel employed in STEM programs at various colleges and universities, having minority faculty members and mentors can be beneficial to URM students. Price (2010) touches on the value of having minority instructors as part of a school's faculty by using data compiled from Ohio public universities from 1998-2002 and finding that black students that had a black teacher during their first semester of college had a 5-8% increase in persistence in their STEM degree program after the first year. However, the effect of black faculty on persistence after that first semester was not significant. Also noteworthy, having a black instructor does not have a negative impact on non-black students. Unfortunately, having a female professor had an opposite effect on students, where female students are 7.4% less likely to persist after the first year and male students are 1.8% less likely to persist (pp. 907-908). If possible, having URM professors or mentors could be beneficial, perhaps as a subconscious referent to students that underrepresented groups can succeed in STEM fields or just as more plainly as a role model.

The Benefit of After-School Programs

In addition to the importance of building an inclusive community of ICT/STEM scholars, students have cited participation in ICT/STEM-related activities (clubs, science fairs, etc.) as important for keeping students interested, particularly because they get to apply what they learned in practical ways (Meador, 2018, pp. 64-68). Many scholars have demonstrated positive outcomes in interest building and long-term preparation from training outside of the classroom in STEM clubs, workshops, summer programs, assistantships, and many other communities of learning that value hands-on experience or tutoring. This study's after-school intervention fits in with existing research on after-school and informal activities efforts to encourage and prepare students to study ICT/STEM. This research project introduced middle school students to computer science content by implementing design theories established by other scholars while providing new insights into practices that can work for a young age group.

Sahin (2013) shows the relationship between participation in extracurricular activities (after-school clubs and science fair competitions) and persistence in studying STEM by looking at student records of after-school activities and matriculation data and declared majors for recent graduates from a STEM oriented charter school system (pp. 6-8). Ultimately, Sahin notes a statistically significant link between students that attended more STEM clubs and declaring a STEM major than students that attended fewer clubs (the study compared 1-3 club memberships). The study did not find a statistically significant relationship between the number of science fair competitions and declared STEM majors, but Sahin notes a pattern showing an increase in declared STEM majors with each additional science fair competition (p. 8). Sahin speculates that the success in science fairs and clubs comes from students seeing how the content they learn applies outside of the classroom, being intrinsically motivated in the after-school activities, or enjoying the creative environment with fewer restrictions (pp. 8-9). Unfortunately, Sahin does not go into detail describing the learning style or objectives of the clubs that students would have participated in (nor could Sahin considering the breadth of the study), but there is nonetheless a strong correlation between frequent exposure to STEM subject matter and persistence in studying.

Other scholars have noted the importance of engagement in authentic STEM activities with the mentorship of subject matter experts in cultivating persistence. Roberts, Jackson, Mohr-Schroeder, Bush, Maiorca, Cavalcanti, Schroeder, Delaney, Putnam, & Cremeans (2018) looked at how participation (particularly aimed at lower SES, underrepresented minorities in STEM fields) in an informal learning environment influenced perceptions of STEM (pp. 3, 5). Their study placed 5th-8th graders in a week-long STEM summer program that was meant to give students hand-on experience in authentic working environments for a variety of STEM fields, with various professors, subject matter experts, and professionals acting as mentors that help students with projects, explain concepts, and just describe their line of work to the students (pp. 4-5, 8).

Students engaged in robotics daily and then worked on smaller projects for other fields like chemistry and biology (pp. 4, 8-11). Data collection included interviews with students and the review of student reflections. The students gave feedback expressing interest in STEM fields, enjoyment of hands-on activities that many did not have opportunities to do in school (lacking equipment, teachers not having time to go into detail or rushing through material, lacking STEM classes, etc.), feelings of preparation for ongoing math/science/STEM classes in school, and realization that what they were learning had application in the workplace, so students were able to make connections between school and work while also seeing professionals as real people and not isolated, nerdy caricatures (pp. 7-11). Students were particularly happy to engage and not just be told everything they need to know (p. 9).

Similar to Roberts et al.'s study, Duran, Höft, Lawson, Medjahed, and Orady (2014) continue the theme of learning in authentic working environments and with professionals in a more prolonged format as they present a study of high school students attending the FI^{3}T after-school project, which featured two cohorts of underrepresented minority, female, and special needs students participating in a two year-long study (2008-2010 and 2009-2011 respectively) in order to gain a better understanding of their attitudes and understandings of IT/STEM skills, technology usage, and careers in IT/STEM fields (pp. 120, 122-123). The program used a "community of designers" approach where students first learned about various STEM fields through presentations and workshop activities and then they narrow down their interest into two subject areas, from which students were put into teams and completed authentic projects while under the supervision and guidance of subject matter experts, members of the university (teachers and undergraduate students), and their local high school teachers. The students learned how to use different

IT toolsets for various STEM fields, conduct research, and create projects/solutions for real problems while completing year-round workshops, externships (field trips to various facilities during the summer program), and seminars, all of which prepared them to create and present projects at a science fair (pp. 119-121, 125). Results of the study were mixed. Students gained experience using IT/STEM tools, won many awards at a science fair for their group projects, and showed positive growth in their attitudes toward STEM content and careers with 55% having either an improved or sustained interest over the course of the study. Some shortfalls in the study have to be addressed, such as why 32% prefer to have a career in other fields and, more importantly, why 13% had a decrease in interest in STEM fields (pp. 125, 129-131). Furthermore, the researchers call for more research to be done to track what students actually do later in life (do they actually go on to pursue college degrees and/or careers in STEM fields?).

Denson, Hailey, Stallworth, and Householder (2015) used a series of 5 focus groups to uncover if informal learning and competitions (occurring as part of the MESA after-school STEM program) are effective at motivating and retaining students studying STEM subjects. They discovered that the MESA after-school program made learning fun: older students liked mentoring younger students and participants developed a sense of camaraderie with their peers. In addition, students improved their time management and organizational skills, had an increase in confidence in their ability to perform in classes, gained exposure to material that they would not be exposed to on their own, and were able to directly apply science and math and see connections with what they learned in formal classes (pp. 11-14). This study furthers the importance of peer collaboration and support, as discussed in the previous section of this report.

Campbell, Skvirsky, Wortis, Thomas, Kawachi, and Hohmann (2014) express the benefit of helping students make connections between course work and what is done in the workforce and making STEM curriculum more culturally relevant to students by focusing on the impact of their activities. They argue that adding a social justice component can add a purpose to what students are learning and providing opportunities for supplemental training can also help by allowing students to find what truly interests them and set a pathway through which they can continue their training (pp. 589-590). This study ties in the importance of making the curriculum culturally relevant to the students.

These studies demonstrate the impact of extracurricular programs on improving student attitudes towards STEM subjects, however, the affordances of every after-school program will vary. The structure of Roberts et al.'s summer program and Duran et al.'s FI^{3}T project involved being with subject matter experts in authentic environments and working hands-on with projects that had application beyond the classroom. Other studies were also more singularly focused on teaching students a specific skillset while they work hands-on with a production suite. For instance, Mouza, Marzocchi, Pan, & Pollock (2016) present an after-school program for 4th-6th graders designed to introduce CT concepts and programming through the use of Scratch, which, similar to App Inventor, consists of a visual programming language (VPL) to help ease novice users into coding in a kid-friendly environment (pp. 86-87). Students were divided into two groups (one with and one without previous Scratch experience) and undergraduate students in computer science (CS) or relevant fields taught each group in pairs, engaging them by first introducing a CS concept then moving on to guided discovery (which generally had group discussion over the CS concept) and project-based activities (pp. 85, 90-92). The researchers looked at a range of data points (from field notes, to pre and post-test surveys, CS assessments, etc.) and ultimately found students gained CS knowledge and showed a positive, though not statistically significant, change in their attitudes towards CS. In addition, there was no significant differences in learning and using CS between both groups and between boys and girls. However, boys were more confident in their skills while girls had more positive attitudes during the pre- and post-test surveys (pp. 93, 96-97). Furthermore, students did make apps for Scratch, with most projects being interactive (71%) and games, though some were not interactive but were, at least in some cases, more like stories (p. 94). So even short-term studies on after-school programs can elicit a positive response and cultivate interest in STEM fields, but it is up to other programs to help further develop the ground that these introductory programs first create.

Similar to Mouza, Marzocchi, Pan, & Pollock (2016), Sáez-López, Román-González, & Vázquez-Cano (2016) also added Scratch into science and art classes from 2013 to 2015 among 5th and 6th grade students in Spain (p. 134). Their study looked at how use of visual programming language (VPL) effected student's motivation/enjoyment, learning processes, attitude towards programming and content creation, and acquisition of programming concepts (p. 133). The researchers used survey and observation data, comparing experimental groups to a control group, to assess student learning and enjoyment (pp. 135, 137). For art history classes, the students demonstrated some learning of class content (a positive value though slightly lower than what researchers desired) and high interest in the subject matter, but their scores for understanding CT concepts and perceived usefulness of the courseware was much higher. Overall, students felt the experience was enjoyable (pp. 137-138). Ultimately, students did acquire course content knowledge while also learning programming, having fun, and seeing utility with what they did.

Though exposure has been shown to aid school achievement, some scholars have revealed the importance of ongoing support if students are to maintain the advantage they gained. Olszewski-Kubilius, Steenbergen-Hu, Thomson, & Rosen (2016) bring to light the necessity of maintaining participation in extracurricular programs in order to continue meeting the needs of students. The researchers introduced a longitudinal study called Project Excite that was created in order to help close the achievement gap between black and Hispanic minority students and their white peers in math, AP/honors class enrollment in high school, and in access to supplemental support (particularly for students with a lower SES status) (pp. 21-22). High potential students from the 3rd to the 8th grade level were able to participate in supplementary after-school, weekend, and/or summer learning opportunities aimed at preparing students for advanced level math, science, and other course or lab work that they would experience in high school as well as build interest in STEM subjects (pp. 22-23, 34). The researchers compared scores for Excite participants and other students in the same school district in Chicago and throughout Illinois (when data was available). Ultimately, Project Excite reduced the math and science achievement gap between high-potential minority and high-achieving majority students, prepared students for high school, and showed a trend in students aiming for more selective colleges with each successive Excite cohort (p. 33). This study manages to give credence to the importance of having extra opportunities available for students to learn school material and get advanced preparation for the kind of topics that they will experience later. It also attests to the usefulness of peer mentoring (since the early grade levels had older students mentor new students) and role models (undergraduates also were available for support of some of the older students). A high school component was also added to Project Excite since the transition into high school is still difficult and students did not always maintain their 3-8th grade achievements without the continued support (p. 35). Project Excite proved that additional exposure to math and science beyond the classroom is essential for preparing students for STEM subject matter.

These studies reveal that collaboration with subject matter experts, working on authentic projects, peer support, and continuous effort to teach students are important for aiding persistence. This study's research project was only developed to be a yearlong implementation to build interest in technology broadly and computer science specifically, but it can serve as a way to build interest, particularly when working in conjunction with other STEM interventions (like robotics) in the broader after-school programs at each site. However, the scope of this study is only in looking at this intervention's ability to build interest in ICT/STEM, teach CT concepts, build self-efficacy, and build interest in continued study. Unfortunately, this intervention did not occur in authentic workplace environments, as we go directly to the schools and offer the program in their media centers and computer labs. We also did not have students directly work with career professionals, but several SMEs made video testimonials describing their careers and offering advice to our participants. Our experts were undergraduate majors in computer science or related fields in addition to the graduate researchers. We also had students create apps that could be used on Android phones, so the issue of authenticity is inherent. Lastly, the theme of student generated apps was ultimately up to them when they worked on the DIY or PBL activities, so they had creative freedom to build apps based on their own interest.

Instructional Design

In a nutshell, the after-school computing intervention's instructional design is a self-directed environment where students choose to join and learn the fundamentals of using App Inventor to create Android phone apps. The curriculum focused primarily on the use of guided activities with step-by-step instructions (called cookbooks) on how to program an app to carry out different tasks. These packets teach students how to use computational thinking concepts to code their apps so that they may perform different functions. Computational thinking "involves breaking down complex problems into more familiar/manageable sub-problems (problem decomposition), using a sequence of steps (algorithms) to solve problems, reviewing how the solution transfers to similar problems (abstraction), and finally determining if a computer can help us more efficiently solve those problems (automation)." (Yadav, Hong, & Stephenson, 2016, pp. 565-566). The CT concepts in the cookbook activities focus mostly on algorithms and showing how the inner workings of a complex app can be assembled and coded piecemeal (problem decomposition), DIY activities give students opportunities to re-apply techniques they have learned into new contexts (abstraction). The DIY activities asked students to either customize their existing apps on their own without step-by-step instructions or create an app of their own from scratch based on a given topic (the LMS had a few themed DIY activities besides the health app) or their own idea. We hoped that students would use the DIY opportunities to create apps that fill some communal need—such as with activity 7, a DIY app where students should create a topic that provides helpful information on a health issue (medical, exercise/fitness, diet, etc.) that could be beneficial to others to have—but the topics were ultimately up to them should they choose to make one from a completely original idea.

Scattered throughout the packets are claim codes that students could use on the after-school intervention's learning management system (LMS) to earn a badge/coin if they either submit an app or answer a quiz question related to topics that are covered in the cookbooks. The coins could be used to purchase prizes and the accompanying badge was placed on a certificate of completion that students received at the end of the semester. The coins add a game-like element to the activities and serve a two-fold purpose of testing student's comprehension of what they are encountering in the packets and giving them rewards for their progress. All of these elements combined to give students a hands-on, educational experience where they are learning through constructing usable artifacts that they could share with others. The next sections of this report will look at some of the literature on the topics of self-directed learning, computational thinking, gamification, culturally relevant education, and culturally responsive computing.

Self-Directed Learning

The after-school computing intervention occurs almost directly after school ends, with students typically having dinner or a snack before attending our workshops. With that in mind, we felt that students should have an experience fundamentally different from the traditional, teacher-centered classroom format. Therefore, we felt that students could be motivated in a self-directed learning, workshop environment where they work on activities at their own pace, have some freedom in choice in how they proceed through the activities, and have mentors that assist rather than teachers that lecture and give direct instruction. Scholarship on self-directed learning (SDL) generally explains the key elements required for structuring a learning environment to be self-directed (Garrison 1997; Harrison 1978; Keller 1979; Keller 1987), but it sometimes advocates giving information for identifying a student's level of self-direction (Grow 1991; Guglielmino, Gugliemino, & Long, 1987; Song & Hill, 2007) or explicitly teaching students how to learn and setup their own benchmarks in order to be entirely self-directed (Grow 1991). All research generally encourages students to become more independent and build self-efficacy.

Grow (1991) uses the Staged Self-Directed Learning (SSDL) Model, which characterizes four stages that both students and teachers can fall into, ranging from a being dependent (stage 1), interested (stage 2), involved (stage 3), or self-directed (stage 4) for students and an authority (stage 1), motivator (stage 2), facilitator (stage 3), or consultant (stage 4) for teachers (p. 129).

The goals of the model are to identify to which level a student belongs, help students move from lower levels to stage 4, and pair the appropriate teaching style to the students (pp. 129, 136-143). Ideally, instructors will be able to switch between teaching styles by providing adequate coaching to stage 1 learners while being able to identify higher level learners so as not to be too hands on and disruptive of their learning style. The instructor should move from acting as an expert to a guide, facilitator, or consultant in a gradual role change throughout the duration of a semester or learning cycle while the student progresses gradually from a dependent to an independent learner. Furthermore, the teacher will need to provide proper tools and motivation to help students become more independent learners (pp. 136-137, 139, 143-144). This model provides guidance for instructors in determining their roles when helping students they notice falling into various levels of dependence, which is important for researchers and mentors as they interacted with students in this study.

Harrison (1978) sees autonomy as the core of SDL, with participants being productive and managing their development on their own, instructors taking on a peripheral role, and participants being encouraged to continue exploring after any formal learning process ends (p. 153). The learner can be motivated to learn based on a problem he or she has (like needing to learn software or a procedure for one's job) as SDL begins with a student diagnosing his or her learning needs. From there, developmental goals are set, educational resources are located, and learning activities are carried out, and all of this can foster the ability to learn in other contexts since this process is meant for ongoing self-development (p. 157). Students that do not control each part of the learning process can still learn from a teacher if the instructor is influential enough and the student does not rebel against the learning context; however, this does not mean anything learned is retained or continued to be used if the student does not find any long-term utility in the acquired knowledge (pp. 159-160). SDL is about moving students towards “autonomy and responsibility as a learner” (p. 161), so the end goal is to help students fall into a pattern of learning so that they can learn on their own in different situations (p. 165). Expecting students to be autonomous all at once can produce anxiety in students that are not used to it, so the learning environment should have a gradual reduction of control over participants by introducing more and more choice, even if the choices are relatively simple, like between choosing to do one of two equal alternative activities (p. 161). Harris particularly found the conflict experienced in more traditional training seminars to be redirected when learners were given choice over what content was learned and by what means with a variety of learning activities rather than forcing everyone into one format (pp. 163-164). Even in a SDL environment, students may still be unresponsive participants despite an instructor’s best efforts (p. 165).

Song and Hill (2007) see SDL as a personal attribute and a learning process, but they are also interested in considering the context, particularly with online classes, and its impact on instruction (pp. 28, 30). Personal attributes include a learner’s motivations, capability of taking responsibility of his or her learning, use of resources and cognitive strategies, and prior knowledge. The learning process refers to the level of learner control, whether a learning environment is heavily teacher or student centered. Context refers to the environment in which learning occurs, including design elements like the types of activities, resources used, time and location of instruction, and available support (teacher, peer, etc.) among other factors (p. 32). A self-directed learning environment is one in which learners control the learning process (planning a project, monitoring their own progress, and evaluating the outcome), use resources and their own strategies, and motivate themselves throughout the process (pp. 32-33). The resources that students have access to can include information resources (like media they encounter online) as well as human resources, like their instructors and peers (p. 33). Song and Hill note the impact of motivation, for procrastination can set in or students may be doing work to meet minimal requirements and not taking full advantages of resources available to them (pp. 34-35).

As a self-directed learning experience, this intervention was designed to pique the interest of students by giving them options (activities, partners, etc.) while allowing them to work at their own pace. If students make it far enough to make a DIY, then they set the standards as far as what it should accomplish and whether they met their goal. This was one of many after-school activities available for students to choose from, so it was essential for the program to provide enough support and interesting content in order to motivate the students to keep returning. The ideal result of SDL is to "equip students to become more self-directed in their learning" and also to become lifelong learners (Grow, 1991, pp. 126-127). This after-school program does not explicitly teach participants about the learning process—needs diagnosis, setting goals, allocating resources, working on the activity (Harrison, 1978, p. 157), and self-assessment of the outcome. However, we engage our participants in ways meant to foster independence and hopefully lead them to become more autonomous learners. A secondary goal of the curriculum for students that persisted, particularly those that worked on DIY exercises, was to develop online research capabilities so that they would be able to continue making apps outside of the program and be able look for additional resources on their own. Introducing students to available resources beyond the after-school program's website arguably would improve their ability to do online research as they look for additional resources and, perhaps, other coding sites to help learn techniques that we did not teach them. So there is some overlap with prevailing objectives that many researchers have in a fully SDL environment, but our implementation of SDL is more embedded in the instructional design rather than explicitly taught in some regards.

This research project also takes the role of mentors very seriously. As Grow (1991) categorizes their role in the SSDL model, our research assistants and mentors mostly served in stages 3 and 4 –advising students on what they should work on next, helping them brainstorm ideas for customizing projects, or occasionally helping to troubleshoot issues—though researchers and mentors occasionally fell into a stage 1 coaching style if students struggled with following instructions early on in the semester (pp. 129, 136-143). We do not directly tell participants to become more self-directed, but we provide appropriate scaffolding along the way and, ideally, students naturally adjusted to using the App Inventor environment and course packets, seek additional resources on their own (or with our prompts) to enhance their apps, and evaluate their DIY activities based on their own criteria for how their apps are supposed to function.

In addition, some scholars have philosophical requirements for SDL that extend beyond the learning style of the students. Merriam, Caffarella, and Baugartner (2007) describe the goals of self-directed learning to be "(1) to enhance the ability of adult learners to be self-directed in their learning, (2) to foster transformational learning as central to self-directed learning, and (3) to promote emancipatory learning and social action as an integral part of self-directed learning." Being self-directed means being able to "plan, carry out, and evaluate" one's own learning (p. 107). Transformational learning entails thinking critically about one's situation, including the "historical, cultural, and biographical reasons for their needs, wants, and interests" (p. 108). The goal of creating emancipated students and pushing for social action includes putting students in charge of as many decisions as possible and giving them easily accessible resources, but it also can extend to having learners study the "sociopolitical assumptions under which they learn and function" (pp. 108-109), which seems to imply that students would be made aware of SDL and take time to explore the social and political ramifications of the act of learning as well as what impact course subject matter can have on their community. This study's after-school curriculum does not have requirements for students to look into the history of programming or learn about underrepresentation of various ethnic minority groups and why it is important for them to close the gap. Our goal is to simply equip students with the ability to create media for mobile phones while using App Inventor so that they can make usable products from their own imagination and hopefully those projects can do some good for a broader community. We are building equity by giving students information about a multimedia platform and how to use it, but the impact they choose to have with their apps is largely up to them (though we can encourage participants to think of building products for a real audience with real needs) and learning about the historical context of ethnic minorities in programming is perhaps outside of the constraints of a 30-60 minute program that meets twice a week though it might be motivating for students to do research on this topic.

Computational Thinking and Programming

The process of assembling the apps, which includes the planning, coding, and the design work students do on the designer screen of App Inventor) is based on the combined use of various computational thinking concepts and practices. Computational thinking is the way one goes about creating a technology-based solution to a problem, which includes analyzing the situation, designing a solution, and building and programming an artifact. Computational thinking does not just boil down to understanding the syntax of a programming language, but rather CT is focused more on the underlying way code is organized and should carry over between programming languages.

Brennan & Resnick (2012) focus on three aspects of computational thinking: "*computational concepts* (the concepts designers employ as they program), *computational practices* (the practices designers develop as they program), and *computational perspectives* (the perspectives designers form about the world around them and about themselves)" (p. 3). Computational concepts would be the computer science concepts that guide how blocks are assembled in the VPL platform. Computational practices explain the process of building apps—building them incrementally, debugging errors, reusing old code, etc. Computational perspectives are the consideration of one's purpose in building an app (creative expression, connecting with others) and the questioning of the purpose or aspects of a technology (pp. 3-11). Brennan & Resnick give strategies for assessing the development of computational thinking in students, including portfolio analysis, interviews based around their projects, and design scenarios where students had to explain lines of code, debug a problem, extend code, or remix a project (pp. 11-19). Finally, the researchers bring up a good question about how fluent students really are with code when they are able to download other projects and simply use their code, so talking to them directly can be a way to really measure what they know (pp. 17-18).

Barr and Stephenson (2011) liken computational thinking to problem solving with the aid of a computer. Students build tools (e.g. phone apps) using CT concepts to process and analyze data and create their artifacts. CT requires students to be able to perform the following tasks:

- "Design solutions to problems (using abstraction, automation, creating algorithms, data collection and analysis);
- Implement designs (programming as appropriate);
- Test and debug;
- Model, run simulations, do systems analysis;
- Reflect on practice and communicating;
- Use the vocabulary;
- Recognize abstractions and move between levels of abstractions;
- Innovation, exploration, and creativity across disciplines;
- Group problem solving; and
- Employ diverse learning strategies." (p. 51)

Students should work hands-on with various tools (like App Inventor) when assembling a product, using trial and error to resolve difficulties and figure things out till they have a working solution. Furthermore, computational thinking knowledge should be transferable and applicable across various subjects (pp. 49, 51). The skills students gain from learning CT should not be limited to use in computer science classes, but students should be able to see their utility in answering questions and resolving problems in other contexts. In addition, anyone studying CT should be confident when dealing with complex issues, be persistent in finding solutions to problems, be able to handle ambiguity and open-ended problems, be willing to collaborate, and aware of one's own strengths and limitations (p. 51).

Repenning, Webb, and Ioannidou (2010) created the Computational Thinking Tools Checklist, which lists conditions that must be met if a CT intervention will have systemic impact. The CT tool must have a low threshold (meaning that it is easy to make a working artifact quickly) and a high ceiling (meaning the created artifacts can be very sophisticated). The curriculum must scaffold flow (becoming gradually more difficult with each activity), enable transfer (the curriculum and tool should be useful in multiple contexts with transferable applicability), and support equity (the tool and curriculum must be accessible across gender and ethnic boundaries). Lastly, the curriculum must be systemic and sustainable in that the tool needs to be usable by all teachers (they should have training) and students should all face the same standards when using the CT tool, which should lead to the tool and curriculum being adapted into a K-12 school system (pp. 266-268). My study's intervention used App Inventor, which similarly has a low threshold and high ceiling as the programming tool (AgentSheets) used by Repenning, Webb, and Ioannidou. The intervention's curriculum demonstrated transfer in that we did not limit the curriculum to game design but also had cookbooks that allowed students to create a range of apps covering different topics, thereby hopefully making it accessible to all students who may not be motivated by any one particular theme (like game design).

Sherman, Martin, Baldwin, and DeFilippo (2014) provide a rubric for assessing computational thinking skills demonstrated in apps. Their rubric contains 14 CT items, but our activities only covered the following 9 concepts: Screen Interface, Naming Components, Events, Procedural Abstraction, Globals with Variables or Text Labels, Loops, Conditionals, Lists, and Accelerometer & Orientation Sensors. These CT items are not all defined in the article, but the definitions can be explained in the following way:

- Screen Interface (the components on the screen and whether they have a graphical interface)
- Naming Components (changing the name of a component from the default to ease later programming efforts)
- Events (actions that are triggered by an event handler)
- Procedural Abstraction (a set of code that operates some part of the app's programming)
- Globals with Variables or Text Labels (data with values that can change; Sherman et al., 2014, p. 5)
- Loops (a procedure that occurs multiple times or an infinite number of times until terminated)
- Conditionals (a procedure that occurs if certain conditions are met)
- Lists (data organized in a sequence)
- Accelerometer & Orientation Sensors (the use of sensors on the phone to trigger events/control parts of the app's functionality; Sherman et al., 2014, p. 6).

The rubric allows apps to have a minimum of 14 points, but we adapted the rubric slightly so that graded projects would have a minimum of 13 points. We did this because every app contains at least 1 Event, so allowing the Events category to have a minimum score of 0, we could see that the app has little to no coding done more easily. Apps with no coding done were not discarded if participants completed most of the design work (the inclusion, renaming, and formatting of most of the cookbook's required components). Sherman, Martin, Baldwin, and Defilippo's rubric is reproduced in appendix A of this report.

Maloney, Peppler, Kafai, Resnick, & Rusk (2008) structure their app building class somewhat more in a similar manner in that the students engage in using CT concepts as they figure out how to use the Scratch platform without being directly taught by the researchers. The students in Maloney et. al.'s study met outside of school at a computer clubhouse used by black/African American and Latino youth age 8-18. The students worked on projects of their own choosing and the researchers did not teach explicitly, but rather just served as mentors. Every few months the group had a "Scratch-a-thon" where members would work on a project for 3-4 hours and then share their result (p. 368). The mentors did not have programming experience, so they primarily model how to learn (like how to find information online), which at times allowed the youth to teach the mentors new concepts with Scratch (p. 369). Ultimately, the researchers looked at over 500 projects by 80 students (boys and girls), and the students showed significant gains in 4 programming concepts while a 5th concept had marginal gains (pp. 369-370). One concept, variables, had to be expressly taught before some of the youth began to use the concept (pp. 370-371). The researchers felt that the use of multimedia got kids interested in technology and is a "promising pathway into programming" with the youth becoming content creators (p. 371).

These studies demonstrate that app building and computational thinking through visual programming language platforms can be a fun experience for children in or outside of a formal classroom environment. The studies all use Scratch or a similar platform whereas we chose to use App Inventor for this study because it gives the added benefit of allowing students to create apps for Android phones that can be shared without needing to be used within the interface of the platform in which it was created (like Scratch).

Papadakis, Kalogiannakis, Orfanakis, & Zaranis (2014) review and compare the Scratch and App Inventor platforms. Both platforms allow “beginner students to focus more on the problem solution and less on the language syntax,” which can speed up app development since they will not have to focus as much on perfectly typing in strings of code (p. 5). Furthermore, students can get right into making the kind of app that appeals to them whereas in “conventional programming teaching…the examples are usually the type of assignments of creating lists of first numbers” (p. 5). For a comparison, the authors point out that App Inventor targets smartphone users, which could incentivize learning programming more than Scratch. App Inventor also has the same features as Scratch with added functionality. The authors also argue that App Inventor’s object-oriented programming environment “makes it easy for the future transition of the user in more elaborate programming languages such as Python or Java through the characteristic Java Bridge.” They conclude that App Inventor “is possibly more appropriate for a more official introduction to programming where the final aim is the strengthening of the programming ability and the transition into a conventional language,” recommending beginning with Scratch for younger users then transitioning into App Inventor (pp. 4-5).

Ultimately, this study teaches students CT concepts without encumbering students with too much jargon. We have them engage in practices that would be familiar to even a seasoned programmer, like storyboarding a concept, remixing old code, debugging errors, and more. We also use App Inventor due to its low threshold and high ceiling, ability to generate apps that students can use directly on their phones, and its stronger semblance (than Scratch) to proprietary platforms that professional app developers may use.

Gamification of Learning

Domínguez, Saenz-De-Navarrete, De-Marcos, Fernández-Sanz, Pagés, and Martínez-Herráiz (2013) describe the pros and cons of gamification when implemented with a treatment group against a control group and students in the treatment group that opted to complete traditional exercises instead (p. 388). They describe the positives of gaming as a means of building interest while engaging students in an active learning process, adding elements of competition (such as through leaderboards), providing a potentially customizable, non-linear learning platform, and giving immediate feedback (pp. 380-381). Games can be motivating because of their complexity and progressive difficulty that leads students to mastery, their effects on emotion when students succeed as well as the anxiety while engaging (provided it does not lead to frustration), and the possibility for social interaction provided that users compete, collaboration, or just communicate (pp. 381-382).

Domínguez et al. created a reward system of trophies and medals to serve as achievements for completing tasks and a leaderboard so students could see their ranking and how many tasks their peers completed while working through an ICT class (pp. 383-384). Students in all three research groups actually received the same assignments, but the experimental group had to upload screen captures of their work instead before they were awarded trophies for completion (p. 385). Ultimately, the control group performed better in the class and final exam, though the experimental group performed better on certain units (pp. 386, 388). The researchers were primarily concerned that the gaming may have helped students "develop practical competences but somehow…hinder the understanding of underlying theoretical concepts in contrast with traditional courseware" (p. 386). However, the experimental group gave positive feedback in their survey responses (p. 388). There was low participation in the gamified version of the course, and most students in the experimental group were less motivated by it. Some students stated that they did not have time for it, felt the system was confusing or experienced technical difficulties, and they did not like the leaderboard/competition. The lack of immediate feedback may have also been a detriment. However, some students liked the competition and colorful/encouraging software (pp. 389-391).

The concept of progressive difficulty is the primary element of gamification that was used in this intervention, as the cookbook activities become progressively more difficult and build upon concepts learned in previous activities. Other elements are optional, such as earning badges and claiming prizes. In addition, our multiple-choice quizzes provide immediate feedback on multiple choice questions and a badge/coin if answered correctly, but app submissions and short answer questions had to be reviewed manually before a coin is released for completion. The prizes that students could earn mainly included school supplies and healthy snacks.

Culturally-Relevant Education

Allowing students to tailor content around their own cultural background is one way that can positively motivate them, particularly with underrepresented ethnic groups that may not have topics directly related to them discussed in school curriculum very often. This enables students to gain a chance to explore aspects of their and other's cultures and issues that have affected them currently and historically. Culturally relevant pedagogy focuses on improving student attitudes towards their coursework by engaging them on a more personal level. Similarly, culturally responsive teaching is another framework that also focuses on helping students become more actively involved in a student-centered classroom by emphasizing cultural issues and striving for social justice. Since there is more cohesion than disconnect between these two learning theories, culturally relevant education (CRE) is the proposed umbrella term to refer to the concept of transformative learning revolving around cultural issues in the classroom (Aronson & Laughter, 2016, pp. 163-167). Ultimately, the emphasis is on creating a student-centered, culturally inclusive environment, and the results should be that students feel more of a personal connection to their coursework, have more understanding about their own culture and respect for that of others, gain more interest in the course material, achieve academic success, and develop an interest that should lead them into lifelong learning and engagement with cultural topics (Aronson & Laughter, 2016, pp. 164; Ladson-Billings, 2014, p. 77; Gay, 2013, p. 51; Choi, 2013, p. 17).

Ladson-Billings, one of the most cited names in CRE, places emphasis on the ability for CRE to increase 1) cultural competence—the ability to gain knowledge of and appreciation for other cultures—and 2) sociopolitical consciousness—the ability to apply what is learned in school to problems that affect people beyond the classroom (2014, p. 75). The biggest benefit is that normally disadvantaged students will now get to tackle topics that may affect them personally or a group that they represent in some capacity. The pedagogical framework of CRE opens the possibility for topics that can be universally appealing due to its interest in critically addressing contemporary issues that affect the world (pp. 76, 78). So essentially, the classroom becomes an exploration of global identities and can even lead to changes in identities over time. In doing this, pedagogy can become culturally revitalizing since it can teach us about the past and cultural features that may be facing extinction while also allowing for pedagogy to become culturally sustaining by looking at and incorporating current practices among students and their cultures in their everyday lives (pp. 82). Ladson-Billings exemplifies this with an example of a seminar where hip hop and performance art were integrated into a teacher training course. She saw a class that would normally consist of predominantly white, female students working with and learning first hand from the critical discourse of their peers enrolled in the First Wave program—

a teacher education program that recruited hip hop artists in the surrounding community to enroll at the University of Wisconsin-Madison (pp. 78, 80).

Choi (2013) similarly highlights the positive influence of a culturally diverse curriculum in the study of a Korean high school instructor in the North-East region of the United States that improved his social studies class by focusing on world religions rather than the standard curriculum focused on the history of western nations. His students, the majority of whom were Asian and Latino, gravitated towards the topic since they were able to discuss their own cultures, participate and even lead discussion in collaborative projects, and ask questions that often led to meaningful discussion about cultures as well as historical and currents issues (pp. 14-16). There was also a critical component to the class since they were discussing the connection between world religions and their positive impacts on societies (in art, music, communities, etc.) as well as the struggles that are going on today as a result, such as between Tibet and China (pp. 16-17).

However, rather than putting too much focus on historical issues, some would rather focus on the contemporary lives of their students. For scholars like Gay (2013), culturally responsive education is about connecting what occurs in the school with what students relate to out of school (p. 49). She characterizes culturally responsive teaching as putting the cultural knowledge, experiences, frames of reference, and performance styles of diverse students into effect in education in order to make learning more relevant to them (pp. 49-50). It is important to be careful that culturally relevant educators not simply couch all the discussions in terms of negatives that have to be overcome. According to Gay (2013),

> Educational innovations motivated by and framed only in negativism do not generate constructive and sustainable achievement transformations for ethnically and culturally

> diverse students. Furthermore, there is an underlying fallacy in the pathological perceptions of communities and students of color that needs to be debunked. This is the assumption of universal marginality, powerlessness, and disadvantage. (p. 54)

So instructors must be careful not to let groups come across as always being disadvantaged and powerless since groups are not always marginalized in all things (p. 54).

This app building intervention falls more in line with Gay (2013) in that we are allowing students to choose topics that appeal to them. In some cases, students might want to make adventure games that would be universally appealing while some students might want to an app that would be more appealing to a targeted group, like fans of hip hop or a civil rights leader. We themed some of our cookbook exercises to appeal to the cultural interests of URM students as Choi (2013) would recommend, but many of our guided activities are relatively neutral in theme (such as with the calculator app or the Space Invaders game app).

Culturally Responsive Computing

Like culturally relevant education, culturally responsive computing attempts to bring student's cultural background to the forefront, but specifically in computer science education. Scott, Sheridan, and Clark (2015) view culturally responsive computing (CRC) as a way of blending culturally responsive pedagogy with technology education in order to make it more accessible to diverse sociocultural groups. Furthermore, the goal is to help students create with technology while considering their social relationships and pushing for community advancement (p. 413). The driving force behind CRC, then, is the placement of more focus on student's culture, which would improve their self-concept/image and sustain their interest by not focusing on the dominant culture and by also having students engage in critique of the social order while tailoring all of this towards STEM learning experiences (pp. 414-415, 417). Scott, Sheridan, and Clark also propose a heavy focus on student's reflection on their intersecting, sociocultural identities so that they consider every variable that makes up their identity—and not just one or two, like ethnicity and gender—in order to show their identities are not exclusive to social class (pp. 421, 423-427). The end goal of CRC is for students to become inventors of technology that serves their communities to some end, with some emphasis on subversion of the status quo (pp. 425-427). Success is then judged not on outcomes or standardized testing but rather on the impact students have on disenfranchised communities (pp. 429-430). Meaningful collaboration and engagement with the community, then, becomes a major focus on CRC (p. 430).

Lachney (2017) backs up Scott, Sheridan, and Clark (2015) but places more emphasis on interaction with the community. Lachney first points out that most culturally responsive education focuses on surface level changes to lesson content—like adding ethnic characters or contexts to problems or scenarios (p. 421)—whereas he recommends deeper reform. Lachney presents culturally responsive computing (CRC) as a way of having educators, software developers, and the broader community work together to add the cultural capital from representative members of the community into the school curriculum so that students are encountering lessons based around themes from their home cultures rather than the mainstream standard (middle class, white American culture). This is done by having the community identify skills kids should have and then negotiate with educators and software developers to create lessons that students can use to learn about their cultural assets (pp. 421-423, 425, 432-434). Going even further, Lachney points out how there must also be a positive impact on the community, for the ends of the education should lead to some productivity (asset building) and advocacy for the community (pp. 421-423).

In their article, Ryoo, Margolis, Lee, Sandoval, and Goode (2013) describe a program, Exploring Computer Science (ECS), which essentially embraces "inquiry-based, hands-on, culturally relevant instruction" and is meant to build student interest in computer science through activities that are meaningful and have a purpose for the student. Furthermore, ECS puts emphasis on student authorship of projects based around topics that meet their own interests (p. 164). ECS, then, is enmeshed with culturally responsive computing though some scholars like Lachney (2017) would necessitate direct involvement with members of the larger community whereas Ryoo, Margolis, Lee, Sandoval, and Goode (2013) put more emphasis on the students bringing their own knowledge, histories, and experiences to the table though they can always consult others (peers, teachers, etc.) for support (p. 165). Furthermore, in their article, the authors put more emphasis on the nuts and bolts of teaching of CS curriculum (computing, web design, programming, data analysis, etc.) to equip students with problem-solving skills as they work on inquiry-based projects (pp. 165-166). The role of teachers may be one of facilitation rather than lecturer, but the ECS program also emphasizes the professional development of instructors so that they are networking with the broader teaching community to enhance their knowledge of CS content, culturally relevant pedagogy, and challenges with stereotype threats that students face (p. 167). Now, the authors do advocate for having students work on projects centered on social issues that will positively affect their local communities (p. 176), but this is would be a unit that follows some general practice and foundation building.

This intervention pairs with Ryoo, Margolis, Lee, Sandoval, and Goode (2013) most closely since we seek to equip students with CS and CT knowledge that they can then use as a starting point to build their own projects on topics that appeal to them. Lachney (2017) puts more emphasis on involvement with the community, which is a layer that could enhance the learning experience if students worked on PBL activities with true clients (parents, other youth groups, charities, churches, senior citizen homes, local fire departments, libraries, etc.). The larger community's involvement in this research project came in the form of mentorship in learning programming/app building rather than as clients for whom students would build a project. Scott, Sheridan, and Clark (2015) also put more emphasis on social justice themes for subverting unjust conditions whereas our long-term emphasis was more on the student's unique interests, whether that was challenging established norms or just creating apps that were fun or helpful to them in some way.

Having discussed the value of this intervention and the theories behind the instructional design, attention shall now be placed upon the analysis of the intervention based on its ability to meet its objectives of teaching computational thinking skills and building interest in STEM. This will be done using a design-based research framework project and grounded theory for creating new theories pertaining to what aspects of the intervention was successful in building interest in STEM and motivating students to continue working.

Design Based Research

Design-based research is about creating an intervention and engaging in cycles of usability testing in order to improve the design and expand understanding of the theoretical framework guiding the creation of the intervention. Joseph (2004) indirectly describes design-based research as a process of "designing an intervention that, through subsequent iterations, gets better and better at activating and supporting that aspect of learning" (p. 235). Furthermore, "design researchers focus on questions that impact the design and even more tightly on questions that address the key hypotheses embedded in those designs" (p. 237).

In a nutshell, design-based research is about creating an intervention that is refined across iterations with researchers asking and attempting to answer questions related to the design and its underlying theories. Beyond the importance of building an intervention through rapid prototyping and testing in iterations, Joseph (2004) notes the importance of building upon prior knowledge to create new instructional theories or fill in gaps in our understanding as well as to help narrow our focus on what aspects of the intervention need further exploration (pp. 235-237). Wang & Hannafin (2005) similarly promote how design-based research emphasizes instructional design, research methods, and practice, with importance given to the research process, participants, context, and design of the intervention, which is tested systematically across each new iteration in order to improve the design and refine its pragmatic and theoretical goals.

The Design-Based Research Collective (2003) also emphasize design-based research as the development of innovative learning strategies in context by giving five characteristics of design-based research: 1) the design of the learning environment is paired with developing a learning theory, 2) development happens cyclically with ongoing revision, 3) emerging theories and relevant implications must be shared with others, 4) research must occur in authentic settings, and 5) appropriate methods should be used to document and analyze the learning theory with the learning process, curricula materials, and outcomes (p. 5). Care must be taken in determining how all parts of the intervention—including the setting, instructional materials, the curriculum, instructors, the learners involved, and scaffolding among other things—interact and lead to observed learning and whether the results can be generalizable beyond that specific context (pp. 5-6). Furthermore, evaluation occurs during and after the intervention, with formative evaluation occurring during iterations in order to gather information and explore ways that the design could be improved while summative evaluation happens at the end in order to refine theories of learning more so than to create a perfect educational artifact (p. 7).

So DBR emphasizes the creation of a new intervention and refining a learning theory as its end goal. Wang and Hannafin (2005) denote this in the first two of their five characteristics of design based research, arguing that it is: **pragmatic** (it leads to refinement of theory and practice); **grounded** (it is grounded and informed by relevant research, theory, and practice); **interactive, iterative, and flexible** (designers work with participants, work in cycles with new iterations, and the design is flexible and can be adjusted as needed); **integrative** (data is collected through mixed methods, depending on the needs at a given time); and **contextual** (the setting and design process are connected) (pp. 7-8). This means that design-based theory does not occur in isolation; rather, a new learning innovation needs to be built upon prior research and add new insight into the broader literature of a field of study. Researchers begin design-based research with a theory about student learning and practical instruction and revise their theories during each iteration of the project based on their research within a specific, authentic context (pp. 8-10). This means that the underlying theory is used to guide the creation of an intervention which is tested with the target audience in an authentic environment that end-users would normally inhabit, which could be a classroom or a true workplace if, for instance, the goal of an intervention is to introduce novices to aspects of where career professionals work.

Wang and Hannafin (2005) introduce the importance of data collection through a variety of methods and sources, such as surveys, participant exams, interviews, and observations, among other means, for the sake of improving the validity of the results and ensuring the intervention is feasible in a real setting for achieving its goals (p. 10). This means that qualitative and quantitative data collection methods can all lead to an improved understanding of what aspects of an intervention contribute to its goal, what part of the theoretical framework can be enhanced, and whether the very research questions need to be changed in order to put more emphasis on different aspects of the study. This can be done through continuous evaluation over the course of the research project. Assessment occurs as formative and summative (or retrospective) evaluation, which takes contextual data from the study in order to make any claims or changes based on the provided evidence. Formative evaluation focuses on the design and identifies issues that the research project should investigate. Retrospective evaluation focuses more on the theory and how any emerging issues can be used to guide future investigation (pp. 10-11). DBR researchers tend to stress that summative evaluation is mainly interested in refinement of a theory since the testing of an intervention is context driven and every context will differ even if you perfect it in one (Wang & Hannafin, 2005, pp. 10-11; The Design-Based Research Collective, 2003, p. 7). If the results need to be more generalizable, then it is important to test in multiple contexts (Wang & Hannafin, 2005, pp. 10-11) and likely at more than one time.

Some researchers focus more on the early and final stages of a design-based research project more so than theory generation, discussing the considerations that have to be made when crafting a learning design and when analyzing the effectiveness of the design after implementation. Bielaczyc (2006) introduces the Social Infrastructure Framework as a means for developing a learning design and analyzing its effectiveness (p. 302). The framework has four dimensions that must be considered when creating an instructional design: cultural beliefs, practices, socio-techno-spatial relations, and interactions with the “outside world” (pp. 303-304). Cultural beliefs revolves around the conceptualization of learning (do students see themselves as generating knowledge or just consuming fixed information?), the students' and teachers' identities (do students see themselves as investigators and are teachers seen as facilitators/co-learners?), and the technology being used (what is the purpose of the technology and how is it being implemented?). The cultural beliefs set the stage for the other three dimensions (pp. 304-306).

The practices dimension addresses issues of activity selection, how tools are taught (teach functionality separately or embedded in other activities), what should students produce, whether they should reflect on their work, whether students should collaborate or not and how that should be organized, instructor participation, and activity coordination (Bielaczyc, 2006, pp. 307-308). The socio-techno-spatial relations dimension is concerned with physical and online support for students, access to technology (is it in the room or a lab that students have to travel to), how work is available and carried out online, the teacher's interactions online, and the interplay of online and offline work (pp. 310-311). Lastly, the interactions with the "outside world" dimension revolves around bringing in outside expertise, sharing student projects with others outside of the classroom (making the work feel more authentic), and interacting with outsiders to exchange ideas (pp. 312-313).

This study is not a true design-based research as it will only cover the second iteration of the instructional design. Nevertheless, this study's underlying interests of curriculum development, use of mixed methods for data collection, assessment of the intervention, and learning theory generation match those of DBR. Principles of grounded theory will later be used in the analysis of our data as the most successful aspects of the instructional design will be noted and used to mold new learning theories or provide support to existing theories. In addition, recommendations for future changes will be made and likely supported by findings from existing research.

Vanderhoven, Schellens, Vanderlinde, & Valcke (2016) provide a model example of design-based research in action. They used design-based research to create and modify an instructional unit on safety when using social networking sites. First, the researchers analyzed the current needs of their situation by observing student behavior on Facebook, evaluating currently available instructional units on online safety, giving students a survey to gain information on their attitudes towards online safety, and creating a framework from existing literature on what instructional materials would be effective for their needs (pp. 460-462). After looking over the results from this phase of the project, the researchers then created a learning unit since everything available was too broad in scope or failed to meet their needs. They tested their instructional unit across five iterations, making changes along the way to get better results (pp. 463, 467-468). The researchers then discuss the changes they made to the instruction each time, the theory in existing literature that informed their design changes, and the results from implementation against a test group and a control group (pp. 469-474). Lastly, the researchers created a derivative learning theory based on the aspects of the instruction that worked (pp. 474-475).

While this article provides a good framework, this research project is much larger in scale than a single, one-hour unit and has more parts to discuss and more data collection methods to review. Therefore, only the 2nd iteration of the study will be reviewed in this report. Analysis will reveal a) what aspects of the instructional design helped us reach our objectives, b) further ideas for ways to improve this project's instructional design based on feedback from all participants, c) any questionable findings that arise that cannot be fully answered from the current data, and d) any other revelations that spark our interest. Grounded theory produces frameworks based on the findings of ongoing research and it is supported by the collected data. That being the case, it is theory building as a work in progress, and theories established by this study should be further fleshed out in future versions of this research project.

Summary

The information provided in this chapter was meant to underscore the importance of building interest in STEM/ICT within largely underrepresented groups (e.g., ethnic minorities and females) for their benefit as they prepare to enter college and the workforce. Most scholarship on building interest in STEM among underrepresented students advocates for early immersion and the use of extracurricular STEM activities, similar to the context for this study. Scholarship in self-directed learning gives advice for making students more motivated and autonomous, which could lead to them becoming lifelong learners if they gravitate towards the subject matter and see utility in it. Computational thinking and culturally relevant education underlie the activities of the intervention, which work to instill/prolong interest in ICT/STEM through app building. This intervention utilizes recommendations from previous research for structuring the class and participant roles as independent learners. Additionally, design-based research and grounded theory were used to guide overall project development and analysis.

3 METHODOLOGY

This chapter will present and defend the study's concurrent triangulation mixed-methods research design for data collection and analysis, which combined data from quantitative and qualitative sources in order to answer the research questions (Creswell, 2005). The study was guided by the following questions:

1. Was there a relationship between participant engagement (i.e., the number of apps completed) and outcomes related to Computational Thinking?
2. Was there a relationship between participant engagement and their reported belief in their 21st Century abilities?
3. Was there a relationship between participant engagement and their opinions about ICT subject matter and/or their desire to persist in ICT?

Data sources included participant surveys, a multiple-choice quiz, student and teacher interviews, researcher fieldnotes, and student artifacts. This broad selection of data sources was aggregated in order to provide a broad overview of participants' daily activities, highlight any change in self-reported interests and motivations, and provide any insight into participants' ability to transfer what they learned from the guided, instructional packets into other contexts, including their own unique projects. See Table 4 for a matrix that maps research questions to data sources in the current study.

Table 4

Research Questions and Data Sources

RQs	Data Sources
1. Was there a relationship between participant engagement (i.e., the number of apps completed) and outcomes related to Computational Thinking?	• Student artifacts • Quiz results • Fieldnotes
2. Was there a relationship between participant engagement and their reported belief in their 21st Century abilities?	• Survey results • Interviews • Fieldnotes
3. Was there a relationship between participant engagement and their opinions about ICT subject matter and/or their desire to persist in ICT?	• Survey results • Interviews • Fieldnotes • Student artifacts

The following sections will include a rationale for the chosen methodology, the context of the study (including the setting and participants), the instruments for data collection and analysis, and the procedures for collecting and analyzing data.

Rationale

Mixed methods research is useful for its blend of quantitative data, which uses a series of closed responses with predetermined choices, and qualitative data, which consists of open-ended responses from participants (Creswell & Creswell, 2005, p. 317). Mixed methods research integrates data by using findings from one method to punctuate the analysis of another or by attempting to quantify the qualitative data and thereby use it in conjunction with quantitative data. Mixed methods data also can be used sequentially, with findings from one methodology determining the direction of further research with the other methodology (p. 318), but that approach is not used in this research project. For this study, data was collected during the same timeframe, and analysis of the quantitative and qualitative data occurred concurrently. In triangulation mixed methods research, much of the data that is collected from one methodology overlaps with the data collected from the other and comparisons can be made with one data method perhaps providing insights into the situation that the other cannot, which makes this research design ideal for this study.

According to Creswell (2005), triangulation mixed methods research design is appropriate when it is desirable to "simultaneously collect both quantitative and qualitative data, merge the data, and use the results to understand a research problem" (p. 514). The advantage of mixed methods is that the strengths of quantitative data makes up for the weaknesses of qualitative data and vice versa (p. 514). For instance, quantitative data collection allows researchers to gather data from a wide range of people more conveniently than qualitative data (e.g. an online survey can reach a vast number of people quickly whereas phone interviews would take much longer to conduct with only a fraction of the population though the interview provides opportunities to ask follow up questions that a static survey cannot). Ultimately, the idea behind mixed methods research is to see if the data from one source complements or contradicts the data from another (p. 520). Integration of the data typically occurs when the data is being interpreted, with the discussion centering on the extent to which the findings converge in order to corroborate the results (Hanson, Creswell, Clark, Petska, & Creswell, 2005, p. 229).

Creswell and Creswell (2005) present nested mixed methods design as a way of putting greater emphasis on either quantitative or qualitative data while using the de-emphasized methodology to simply provide support for the other. Furthermore, nested mixed methods research designates different research questions and hypotheses to each data collection method. For instance, quantitative data can be used to "understand the impact of an intervention on outcomes" while qualitative data can help us "understand the process that participants undergo during the study" (pp. 320-321). Hanson, Creswell, Clark, Petska, and Creswell (2005) echo this idea, stating that each methodology is used to answer different questions in order to "gain a broader perspective on the topic" (p. 229). This gives researchers a different area of emphasis for the qualitative data and the quantitative data. This study used a concurrent triangulation mixed methods approach to data collection and analysis, but some data collected from the qualitative and quantitative methods did not overlap, in which case, a nested mixed methods research approach to analysis was used to an extent, with data from one methodology being emphasized and only tangentially supported by any relevant data from the other methodology.

Context of the Study

This study took place during an after-school computing enrichment program at 9 middle school sites in an urban school district in the southeastern United States during the 2017-2018 academic year. This after-school intervention was meant to be a year-long program (encompassing both the fall and spring semesters), although participants were allowed to join at any point during that time. This study focuses on the second major iteration of the after-school intervention (the first iteration, a pilot study, occurred at one site during the previous year).

Setting

While the intervention that is the basis of this study took place at 9 middle schools located across a large urban school district in the southeastern United States, it was embedded in a larger after-school program that served around 2600 students at the time the study was conducted. Sessions typically were held twice a week between Monday and Thursday for 45 minutes to 1 hour. During the spring 2018 semester, public schools were closed for 1 week in January due to snow conditions, which caused after-school programs to meet at a delayed time for the majority of the 2nd semester. This time delay cut down time for all programs and affected attendance for many students.

The intervention took place in computer labs or library/media centers at each school, and graduate research assistants partnered with a member of each school's faculty to help deliver the intervention at each site. Undergraduate students studying computer science or related fields at local HBCUs were also brought in to serve as mentors at each site. While intervention activities were available on a dedicated website, activities were also available to participants as a hard copy. Computers with an internet connection and mobile phones (provided by the researchers) were the only essential tools needed for this study.

Participants

The target audience for this study was a group of middle school students participating in a large after-school program in an urban school district. There were around 2600 students who were enrolled in the program at the time the study took place. Inclusion criteria for participants included students from the given program who both self-selected into the computer science intervention and assented to participating in the study. While 323 students enrolled in the program at various times, only 202 assented to participating in the study. Out of the participants that assented to being in the study, only 120 students actively participated in our data collection efforts. Of these participants, the known ages ranged from 10 to 16 with a mean age of 12. This group included 53 girls and 40 boys, with the majority of students being black/African-American, Hispanic/Latino, or mixed ancestry. Any middle school student attending the larger after-school program could join or leave the intervention at any point during the school year. This choice was meant to make the curriculum and intervention as accessible as possible for the target population, but it provided significant challenges for data collection and, ultimately, analysis as participants were not easily accessible for completing pre-test and post-test surveys and quizzes. See Table 5 for student demographics.

Table 5

Student Tally and Self-Reported Student Demographics for the Main Rollout

School	Students w/Assent	Total Students	Average Age Reported	Gender	Participant's Reported Gender	Ethnicity	Participant's Reported Ethnicity
100	31	43	12.63	Male	19	Black	15
						Mixed	3
						Latino	1
200	23	44	11.58	Male	2	Black	2
				Female	4	Black	3
						Mixed	1
300	28	47	12.27	Male	5	Black	4
						Mixed	1
				Female	6	Black	5
						Native American	1
400	17	34	12.64	Female	14	Black	12
						Mixed	2
500	24	40	13.39	Female	9	Black	3
						Mixed	6
600	9	16	11.20	Male	4	Black	4
				Female	1	Black	1
700	29	34	11.57	Male	7	Black	6
						Mixed	1
				Female	6	Black	4
						Mixed	2
				NA	1	White	1
800	18	29	12.25	Male	2	Black	2
				Female	3	Black	1
						Mixed	1
						Native American	1
900	23	36	11.64	Male	1	Black	1
				Female	10	Black	9
						NA	1

Data collection

Quantitative data, which included CT exam scores, profile survey responses, and app data (scores and number completed), was collected to look at student outcomes in gaining knowledge about CT concepts, developing (or maintaining) a positive opinion towards ITC content and their ability to use 21st century skills, and building an interest to continue learning in this field. Qualitative data was collected simultaneously with the quantitative data partly for the sake of triangulation as the data addresses many of the same questions while also focusing more on the student's learning process. Changes in opinion, self-efficacy, and desire to persist should align with findings from the profile surveys.

The profile surveys and CT exams were completed by participants via Qualtrics, an online survey platform. They were administered after each new participant's first week of class. We postponed delivery of the pre-test so that participants would have a chance to experience the after-school intervention and not confuse our regular activities with the data collection protocols. This was also done to prevent early attrition in case students were reluctant to engage in data collection but might enjoy the standard curriculum. The post-test activities were administered towards the final weeks of each semester, with researchers asking participants to complete it whenever they were present or, in some cases, requesting after-school program coordinators to have them return for the sake of completing the surveys (though this was not always possible).

Not all participants joined the intervention at the same time, so some participants may have missed the timeframe set aside for pre-testing (which was cut off after the mid-point of the semester). Therefore, the very first survey and CT exam that a participant completed was counted as their pre-test and their last survey and CT exam was counted as their post-test, and only participants that completed at least two of each were used in analysis for most sections of this report. Additionally, there cannot be a pre-test and post-test comparison across each semester because participants returning for the spring semester were told to skip the spring pre-test since they would have completed the fall semester post-test less than a month prior, before the Christmas holiday. Constant testing was looked upon very unfavorably by many of the participants, so omitting the pre-test for returning participants was meant to cut down on the workload and curb attrition. Lastly, to encourage involvement in the testing and interviewing, rewards were offered to students for contributing. Participants that completed the post-test survey and quiz and submitted 2 or more apps were eligible.

Table 6 shows the relationships between the topics under study in this report and the data sources. Qualitative and quantitative data may be useful for answering each of our research questions, with triangulation feasible where data from different research methods overlaps. Lastly, only students that submitted apps will be included in this study's quiz, interview, and survey analysis, but all students will be considered in analysis of the fieldnotes.

Table 6

Concepts Being Studied and Their Respective Data Sources

Concept Under Study	Data Sources
CT Knowledge	• App Completion Numbers and Scores • CT Scores • Fieldnotes
Self-Efficacy	• Profile Survey/ICT-21Q • Interviews • Fieldnotes
ICT Interest and Persistence	• Profile Survey/SSS • Interviews • Fieldnotes

CT Quiz

The CT quiz (Ayer, Cohen, & Calandra, 2018) included 12 multiple choice questions related to some of the CT concepts that they would have encountered in the guided app building activities, including events (parallel or sequential), global variables, operations, loops, and conditionals. The CT quiz heavily favored CT concepts that were taught in the more advanced activities (cookbooks 6 and up). Theoretically, students that completed the most work are the ones most likely to have succeeded on the post-test exams, though some students may have completed fewer projects but still skipped ahead and worked on the more difficult projects. A similar theory is that students that were exposed to specific CT concepts more frequently will have higher scores on the CT quiz post-test. Both of these theories will be analyzed in chapter 4. Table 7 shows the relationship between the CT quiz questions, the CT concept most relevant to the question, and the cookbooks that most directly cover the CT concept.

Table 7

CT Quiz Questions and Their Corresponding CT Concepts and Cookbooks

CT Quiz Question	CT Concept	Relevant Cookbooks
1	Loop	6, 9, 10
2	Event (Sequence)	2, 8, 9, 10, 11
3	Variable	8, 9, 10, 11
4	Variable	8, 10, 11
5	Event (Sequence)	2, 8, 9, 10, 11
6	Event (Sequence)	2, 8, 9, 10, 11
7	Event (Parallelism)	2, 6, 8, 9, 10, 11
8	Event (Operation)	6, 8, 9, 10, 11
9	Event (Sequence)	2
10	Loop	8, 10, 11
11	Conditional	6, 9, 10
12	Conditional	6, 9, 10

The CT quiz questions show images of blocks of code, and students were asked to decipher the code by explaining its function, figuring out how to complete or change elements of the code, or identifying an error (see figure 6). The quiz did not change from pre-test to post-test. While it is possible that students may remember seeing the question before, we did not give them the correct answer upon completing the quiz. Therefore, they had to comprehend the CT concept in order to do well on the quiz.

Q1 Look at the code below. What do you think happens once the score gets to 2?

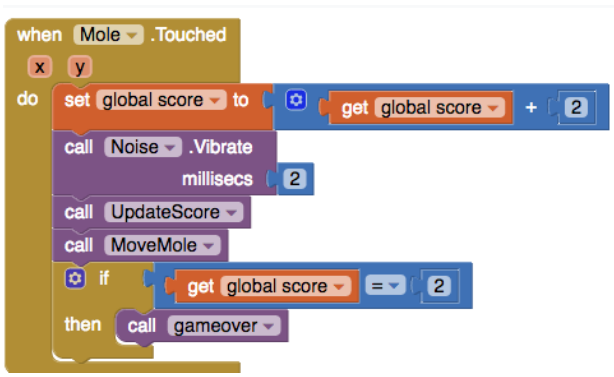

Figure 6: Sample CT quiz question

Profile Survey

The profile survey data included questions that requested information about each participant's background with programming, interest in ICT/STEM subject matter, satisfaction with the after-school program, and more. Two important parts of the profile survey were the STEM Semantic Survey (SSS) and the ICT/Twenty-First Century Skills Questionnaire (ICT-21Q). The SSS was developed by Tyler-Wood, Knezek, & Christensen (2010), and it was used to gauge one's change in interest in ICT/STEM subject matter from pre-test to post-test. The ICT-21Q was developed by Cohen, Renken, and Calandra (2017) to gauge the student's self-concept of their ability to use 12 different skills valued by ICT professionals, including some that are technology focused—like the ability to work with technology systems—and some that are not—like problem solving and critical thinking (pp. 383-384). For this report, only data from the STEM

Semantic Survey pertaining to the student's opinion on technology and careers in STEM fields is reported as these questions directly pertain to their interest in ICT while questions about math, science, and engineering are arguably outside of the scope of interest for this study.

Beyond the SSS and ICT-21Q, various multiple choice and short answer were reviewed since they probe for background information on the participant's experience with coding, the students' interest in continuing to code, and feedback on the after-school intervention. Multiple choice questions pertaining to the student's background and confidence include the following:

1. I feel confident in my ability to learn how to build phone apps (with responses falling on a Likert scale, 1-5, from "Strongly Disagree" to "Strongly Agree").
2. Have you built an app before (with responses standardized to "yes" or "no")?
3. Have you ever tried to write computer programs or code before?

Short answer questions pertaining to motivation and confidence include the following:

1. Why did you choose to join the intervention?
2. Were there any times when you felt good at what you were doing?

Multiple choice questions about ICT interest and one's desire to persist in learning about app building include the following:

1. Which sentence best describes how you feel about coding?
2. How much did you like what you were doing (Likert Scale, 1-4, low to high)?
3. Would you like to continue with activities like this intervention?
4. Will you create apps on your own outside of the after-school program?

Short answer questions about ICT interest include the following:

1. What was your favorite part of the intervention?
2. What was your least favorite part of the intervention?

3. What would make these activities more fun?

Interviews

Semi-structured interviews (Creswell, 2005) were conducted with as many students as possible at the end of each semester, after the survey and CT quiz were completed. Researchers asked leading questions and typed the participant's responses directly into a Qualtrics form. Probing questions were used to help generate responses, but time constraints prevented inter-views from going too in-depth and interviews were cut short in some instances. Some of the questions included having participants describe times they felt good about what they were doing in the after-school intervention, what they liked the most and least about the program, their thoughts about their own development of real apps, and why they chose to attend. Many of these questions were triangulated with the survey data. On the other hand, teacher interviews were only held during the mid-point of the spring semester. These interviews were not captured online but via pen and paper or Microsoft Word. See appendix C for a list of the most relevant student interview questions used in this report.

Fieldnotes

All researchers recorded daily reflections online via a Qualtrics survey at the end of each session, and some researchers wrote more detailed fieldnotes concurrent to or directly after meet-ings of the after-school intervention. The online reflections ask general questions about the suc-cesses and difficulties experienced during the day's meeting (such as, whether students worked in teams, if anyone struggled with our technology, or if any aspect of our intervention seemed especially helpful during the period). The fieldnotes dove deeper by allowing researchers to give as complete and detailed an account of what happened in the classroom, with the limitations just being what the researcher was able to observe and what he or she can recall at the time of writing. The fieldnotes received the focus in this report with the reflections adding additional information on occasion.

App Projects

App completion numbers, scores, and grader notes constitute the final source of quantitative data. Scores and assessment standards for the participant apps were compiled by Hicks and another researcher using a rubric derived from Sherman, Martin, Baldwin, and DeFilippo (2014). The apps were not graded as one would traditionally score a student assignment by deducting points for mistakes based on how closely they matched their app to a model. Rather, points were awarded based on the participant's demonstrated usage of 14 various CT skills, including the use of events, global variables, conditionals, and loops. We followed the Sherman, Martin, Baldwin, and DeFilippo rubric with one exception: we awarded 0 points for the Events/Eventhandlers rubric item if the student did not have any code, thereby giving each app project a baseline score of 13 points. Every app has at least 1 event, so a score of 0 for that CT concept made it easier to immediately notice if a student did not have any coding completed. Incomplete projects were not discarded if participants worked on adding and formatting the components since that illustrates work done on the design of an app. Furthermore, our most complex guided activity had a maximum score of 27 points (with our shortest activity having a ceiling of 15 points), but student DIY projects had a potential maximum score of 45 based on the point distribution of the rubric. However, to reach closer to 45 points, students would have to learn CT concepts and features of App Inventor that we do not cover in our activity packets. Sherman, Martin, Baldwin, and Defilippo's rubric is reproduced in appendix A of this report.

For each of the guided app activities, the two graders agreed upon a maximum point value for each cookbook activity and how points should be distributed among the 14 CT skills. Students that modified their app and went beyond the requirements of the assignment had their app graded as a DIY project instead, which makes them eligible for additional points depending on the amount of additional work that was done. If the customizations were surface level (like simply adding an additional background color) and would not have received additional points based on the rubric, then those apps were not considered DIY but they received additional notes in our gradebook to indicate the modifications. Hicks graded all of the projects, so only his scores were used in this report.

Data analysis

The quantitative and qualitative data was used in conjunction with the number of apps students made in order to determine if participants who were engaged in more work a) learned computational thinking and performed better on the CT exam, b) gained a higher interest in ICT and saw value in what they were doing, c) felt more confident in their ability to code and use other 21st century skills, and d) desired to continue learning ICT content. These concepts are addressed by grouping students into quartiles according to the number of apps they completed:

- Quartile 1 (1-2 apps),
- Quartile 2 (3-4 apps),
- Quartile 3 (5-6 apps), and
- Quartile 4 (7+ apps).

Comparisons between quartiles for each data set are presented only if students contributed to pre- and post-tests for the SSS and ICT-21Q portions of the profile surveys and the CT exams. Measurements for these items specifically focus on the post-test scores and changes in scores from pre-test to post-test. Other survey data, along with the interviews and fieldnotes, more generally looks at the cumulative findings for the intervention though still considering the groupings of the students in their quartiles when considering the interview and additional survey data. Overlapping interview and survey data is compared while fieldnotes are used to fill in gaps or help explain some trends in the data. The fieldnotes also provide supporting or contradictory evidence to go with the other findings. Analysis arguably will reveal that students that completed more work will likely have more positive opinions about ICT content and their ability to perform and succeed in this field of study, and they should also have higher changes in their CT exam scores from pre- to post-test.

When exploring the impact of the completion of relevant apps on the CT quiz scores, students were again grouped into quartiles based on the number of apps they completed that covered material directly relevant to the CT quiz. The groupings for these quartiles are as follows:

- Quartile 1 (1 app),
- Quartile 2 (2 apps),
- Quartile 3 (3 apps), and
- Quartile 4 (4-6 apps).

This change in the quartiles is due to the CT quiz only matching material from six of the cookbook activities, thus giving students fewer chances of exposure to the most relevant content. Students that are not exposed to those six cookbooks may still be able to read the code and understand what the questions are asking, but exposure to the six cookbooks should make the code in the quiz questions appear more familiar to the students. This alternative grouping of the quartiles was only used for one data set in chapter 4 (see the *CT Quiz* section) while the normal grouping of quartiles based on apps submitted was used for all other sections of this report.

Lastly, some data was analyzed both qualitatively and quantitatively. Some interview data and survey responses to short answer questions was coded and analyzed for qualitative analysis while also being quantified based on whether the responses were positive, negative, or neutral and then compared between quartiles using correlation and ANOVA measurements. This procedure made it possible to determine if there are statistically significant differences between the quartiles for some of the major topics under study in this report that are not covered by the CT quiz, ICT-21Q, and SSS quantitative measures, but emphasis is also given to the specific content of the responses and not just the group differences.

Quantitative Data Analysis

Data from the pre-test and post-test of the CT exam and profile surveys was compiled into SPSS along with some of the data from the student interviews. This allowed each question to be analyzed by exploring if there are correlations between the number of apps submitted and pre/post-test scores as dependent variables. Similarly, differences between each group was determined through analysis of variance (ANOVA), with quartile as a factor and the pre/post-test scores as the dependent variable. Post hoc exams for ANOVA measurements (Fischer's LSD, Tukey HSD, and Bonferroni correction) were used to determine if there is statistical significance between the different mean scores of each item of measurement for each quartile of participants. Specific short answer responses from the profile surveys and interviews were also be incorporated into the quantitative analysis. For all measurements of significance, an alpha of .05 (5%) is used throughout this study. Alpha (the level of significance) indicates the likelihood of a null hypothesis (that the intervention does not have an effect on the population) being rejected. If the *p* value (probability) is less than or equal to the alpha, then the null hypothesis can be rejected, which indicates that the intervention had an effect on the participant's performance or responses in our data collection efforts. An alpha of .05 is a commonly used value in decision making towards whether the null hypothesis should be rejected or not (Minium, Clarke, & Coladarci, 1999, pp. 204-207).

Correlation

According to Minium, Clarke, and Coladarci (1999) correlations in statistical analysis can be performed to tell if there is an association between two variables simultaneously. The correlation coefficient can be used to measure the "degree of linear association" between two variables, meaning that scores for two variables with a high degree of correlation will similarly rise and fall above and below their respective mean in a similar pattern (pp. 103, 110), Looking for correlation between independent variables is one important way to show if there is a statistical relationship between them. Calculating the Pearson correlation coefficient (Pearson's r) provides a standardized power and magnitude for the relationship, with scores of ±1 showing a perfect relationship while scores closer to 0 showing no relationship (pp. 115-116).

Along with the Pearson correlation coefficient scores, the Spearman rank correlation (Spearman's rho, r_s) was also included in the results. Pearson's r assumes a normal bivariate distribution, whereas Spearman's r_s accounts for data that does not fit into a normal distribution by converting the data sets into ranks before making a comparison (Minium, Clarke, & Coladarci, 1999, pp. 417-418). This measure was included as a precaution in case data sets are heavily skewed and do not fall into a standard bell-shaped curve. Ultimately, Pearson's r and Spearman's r_s help to show if there is a statistically significant relationship between app completion and knowledge transfer, interest in STEM, and self-efficacy in terms of one's ability to use 21st century technical skills.

Analysis of Variance (ANOVA)

Should the data show a correlation between the number of apps completed and student results for any data point, then it is also useful to include one-way analysis of variance (ANOVA), which compares the means for multiple independent groups (Minium, Clarke, & Coladarci, 1999, p. 324). This allows for students to have their average CT, SSS, and other scores compared across and within each grouping of students in order to succinctly see which groups had changes in their scores and determine if the group differences are statistically significant. Between-groups variation considers any inherent variation from the students in each group plus differences in the treatment conditions whereas within-groups variation only considers the inherent differences between group members (pp. 326-327). The *F* statistic, which considers the variance between groups by the variance within groups, should be above 1 (p. 334), otherwise all of the groups are equal and there is either some problem with our tools for assessing the students or the population all equally understand CT and have similar responses on the surveys concepts despite differences in app completion. *F* scores below 1 in this study's context might be possible if students have different previous experience.

According to Glen (n.d.), the null hypotheses, that all groups have an equal means (and therefore are not significantly different), can be rejected if *p* value is smaller than the alpha (.05). The p value has to be considered as it determines the "probability of getting a result at least as extreme as the one that was actually observed, given that the null hypothesis is true." The *F* value, when compared to the *F* critical value, simply determines if something is significant while the p value determines if all of the results are significant. SPSS will be used to calculate the p value and the *F* value. The *F* critical value (also known as the F statistic) can be found using tables or online calculators like the *Free Statistics Calculators* (Soper, n.d.). The calculated *F* value has to be compared to the *F* critical value, with an *F* value larger than the *F* critical value indicating some significance. The p value being smaller than .05 combined with the *F* value being larger than the *F* critical value allows for the null hypothesis to be rejected. Ultimately, ANOVA can determine if there is a significant difference between the means for each group.

With the groups being compared, there is a risk of unequal variances in the means of the collected data, which is a problem known as Behrens-Fisher (Shingala, & Rajyaguru, 2015, pp. 22-23). In the instances where the groups being compared for ANOVA have very unequal sample sizes, an effect on the variance may be possible with Type 1 error rates being either too liberal (when larger groups have smaller variance) or conservative (when larger groups have larger variance). In the case where there are unequal variances between groups, Welch's *W* ANOVA test gives better results than the *F*-test since the Type 1 and Type 2 error rates are closer to the expected levels (Delacre, Leys, Mora, & Lakens, 2019, pp. 1-3, 7). The Welch test is not perfect for controlling Type 1 errors when skewness is present in the results and the sample sizes are small nor when sample sizes are less than 50 per group (p. 10). Nevertheless, Levene's test of equality of variances will be used to determine if variances are present between groups and if the Welch test should be implemented (Gastwirth, Gel, & Miao, 2009, pp. 344-345).

In addition to the *F*-test, *W*-test (when appropriate), and Levene's test data from a standard one-way ANOVA run in SPSS, a univariate General Linear Model (GLM) will be run in SPSS in order to obtain the power and effect size (Partial Eta Squared) for the results. GLM works with balanced or unbalanced designs (IBM, n.d.), with unbalanced designs indicating that there are an unequal number of subjects in each cell (Minium, Clarke, Coladarci, 1999, p. 362). Results for a GLM from SPSS will simply verify the results of a standard one-way ANOVA. The power of a test tells us the probability that a difference exists between groups; the likelihood that we will obtain results that verify rejecting the null hypothesis. The effect size is the magnitude of the difference between the means for each group (Minium, Clarke, Coladarci, 1999, pp. 309, 311).

There are then post hoc tests that can be used to determine which specific groups differ and explain why the null hypothesis was rejected (Homack, 2001, p. 9). This study will include data for the Fischer's LSD, Tukey HSD, Bonferroni, and Games-Howell tests in order to determine statistically significant difference between groups of participants in terms of their pre- and post-test scores as well as their score change.

Fischer's least significant difference (LSD) can be used if the F statistic is significant, for that tells us that not all treatment means are identical (Pizarro, Guerrero, & Galindo, 2002, p. 161). The LSD test then looks for the smallest significant difference between two means and notes any larger difference as being significant (Williams & Abdi, 2010, p. 1). According to Pizarro, Guerrero, and Galindo (2002), many experts "believe that procedures like Fischer's protected LSD procedure should not be used since they do not control the overall confidence level nor the experimentwise error rate….LSD is the least conservative" post hoc test (p. 162). Furthermore, this method is undesirable for a large number of groups because the LSD test only controls for probability of false rejection for each pair and not the overall probability of some false rejection (p. 161). This is because the α level is not corrected for multiple comparisons, which inflates the possibility for Type 1 error, but it, nevertheless, allows LSD to have more power than other post-hoc tests (Williams & Abdi, 2010, p. 3).

The Tukey HSD test is a more conservative post hoc test and it has strong control over Type 1 errors. Type 1 error occurs when a null hypothesis is rejected when it is actually true (Homack, 2001, pp. 5, 10). Tukey HSD uses the studentized range distribution (q) to "determine the minimum difference between the largest and the smallest means in a set of <u>K</u> sample means that is necessary to reject the hypothesis that the corresponding population means is equal" (pp.

10-11). It is used when a comparison between all pairs of means is needed and preferred when a large number of groups are used (Pizzaro, Guerrero, & Galindo, 2002, p. 162).

Bonferroni correction is the most conservative of the three post hoc tests discussed so far (Pizarro, Guerrero, & Galindo, 2002, p. 162). It is used to reduce Type 1 errors when carrying out multiple comparisons, but it suffers from a reduction in power with each test and increases the possibility of Type 2 errors (Nakagawa, 2004, pp. 1044-1045). Type 2 errors occur when a false null hypothesis is retained (Minium, Clarke, Coladarci, 1999, p. 208).

The Games-Howell test will be used as a post hoc test when there are unequal variances between the groups (Shingala, & Rajyaguru, 2015, p. 22). This data will take the place of the Tukey, LSD, and Bonferroni tests if the Welch test is used and significance is found. The Games-Howell test is based on the Welch test and is appropriate if there is unequal variance and sample sizes, but there must be a sample size of 6 per group (pp. 23-24).

Tukey HSD might be sufficient on its own for this study, but the results for multiple post hoc exams will be compiled for cross reference. The post hoc data confirms or denies that students that were more active in completing apps and collecting coins a) learned CT concepts and demonstrated transfer on the CT exam, b) became more interested in STEM, and c) felt more confident in their ability to use 21st century computing skills. Additional support for these claims and attempts to explain any unexpected findings will be pulled from the qualitative data.

Qualitative Data Analysis

Qualitative data mainly comes from the interviews, fieldnotes, and some of the profile survey questions. While much of the qualitative data is used in conjunction with the quantitative data, some deeper exploration into the aspects of the instructional design that aided participants in their learning of computation thinking concepts, development of self-efficacy, and growth in their opinions towards ICT content may be uncovered in these data sets. The fieldnotes were reviewed on NVIVO with a codebook in order to help determine how the content fits into our major themes:

- **affective domain** (motivations, personal relevance, confidence, student growth, etc.),
- **barriers and conflicts** (attrition, complaints, etc.),
- **instructional design** (coins, incentives, scaffolding, support, etc.),
- **social learning** (collaboration, competition, mentor support, etc.),
- **socio-cultural norms** (anything related to culture, family), and
- **student activity** (what they were working on, customization, showing off apps, etc.).

These themes initially were created by Hicks after open coding the fieldnotes. Hicks and another member of the research team refined the codebook after reviewing representative samples of the fieldnotes and online summaries (in addition to fieldnotes, researchers were asked to summarize their daily activities on an online survey) in order to reach interrater agreement. After reaching agreement, Hicks finished coding the remainder of the fieldnotes and summaries. Aside from the fieldnotes and summaries, qualitative data from the interviews and profile survey short answer responses were more simply organized by quartile with the responses and the number and percentage of students in each group that gave similar answers. See Appendix B for the entire codebook.

Current scholarship indicates that mentorship, peer collaboration, and culturally relevant and authentic tasks among many other instructional design strategies can help build and maintain interest in STEM subjects among URM students (Campbell, Skvirsky, Wortis, Thomas, Kawachi, & Hohmann, 2014; Joseph, 2004; Palmer, Maramba, & Daney, 2011). The researcher fieldnotes for this after-school research study document the daily activities of our students in order to identify aspects of our intervention that cultivated learning and thereby add to the existing literature by either supporting other scholar's claims or identifying new design aspects worth considering for future research. NVIVO helped to identify trends in our data sets, which aided in answering this study's research questions. This qualitative data ultimately was used to attempt to identify the factors that a) caused students in the highest quartile to persist and complete as much work as they did and b) led to changes in their confidence and opinions toward ICT subject matter. The data also helped determine the factors that led to attrition, specific opinions students held about the after-school intervention, and whether they desired to continue studying ICT or app building among other topics.

Summary

In this study, a mixed methods approach to data collection and analysis was employed using the following data sources: researcher fieldnotes, student and teacher interviews, graded student apps, profile surveys, and CT exam assessment results. Fieldnotes were analyzed using a codebook created after several rounds of open and thematic coding in which the codebook was constantly being compared and refined (see Appendix B). Other qualitative data mainly organize the responses to questions by quartile and show the differences in opinions between the groups. Quantitative data were analyzed through correlation, analysis of variance, Fischer's LSD, Tukey HSD, and Bonferroni correction on SPSS. Participants were sorted into 4 groups based on the number of apps completed. CT exam scores and scores for some of the profile survey items were compared across each group to see whether completion of apps led to higher group averages.

4 RESULTS

In keeping with the project goals and objectives, the researcher's expectation was that participants who completed more apps should have more knowledge of computational thinking concepts, higher self-efficacy related to 21st century abilities, more positive opinions of ICT, and a greater motivation to persist with STEM/ICT than those who participated to a lesser extent. Data collection and analysis was driven by the following research questions:

1. Was there a relationship between participant engagement (i.e., the number of apps completed) and outcomes related to Computational Thinking?
2. Was there a relationship between participant engagement and their reported belief in their 21st Century abilities?
3. Was there a relationship between participant engagement and their opinions about ICT subject matter and/or their desire to persist in ICT?

Data sources included a multiple-choice quiz, participant surveys, student and teacher interviews, researcher field notes and reflections, and student artifacts. This broad selection of data sources was aggregated in order to provide a broad overview of participants' daily activities, highlight any change in self-reported interests and motivations, and provide any insight into participants' ability to transfer what they learned from the guided, instructional packets into other contexts, including their own unique projects.

In this chapter, data were organized based upon this study's three research questions. The first major section gives the results of a computational thinking quiz in relation to the student's submitted app projects. The second major section provides information about the student's self-reported confidence in their ability to use 21st century skills from the ICT-21Q as well as related data from surveys, interviews, and fieldnotes that disclose information about the student's self-efficacy and the areas of the intervention that may have helped to build the student's confidence as they constructed apps. The third major section focuses upon the interest of the students towards ICT/STEM content and the intervention while also presenting information on their desire to continue learning ICT content. See Table 8 for a matrix that maps research questions to data sources in the current study.

Table 8

Research Questions and Data Sources

	RQs	**Data Sources**
1.	Was there a relationship between participant engagement and outcomes related to Computational Thinking?	• Student artifacts • Quiz results • Field notes
2.	Was there a relationship between participant engagement (i.e., the number of apps completed) and participants' reported belief in their 21st Century abilities?	• Survey results • Interviews • Field notes
3.	Was there a relationship between participant engagement (i.e., the number of apps completed) and their opinions about ICT subject matter and/or their desire to persist in ICT?	• Survey results • Interviews • Field notes

In order to answer the research questions adequately, participants were placed into 4 quartiles based on level of participation in the program. This was measured by the number of activities they submitted to the grant team by the end of the school year (see table 9). The apps submitted are the only tangible evidence of engagement since a) any observation data is limited to the days that a researcher was present at a school, which did not occur every period that the intervention was held, b) attendance was not consistently recorded by the teachers assigned at each school (nor would attendance indicate if students were actively working throughout the period), and c) there were no other deliverables from students to indicate their ongoing activities except their apps.

Table 9

Participants' Groupings into Quartiles Based on App Submissions

				Reported Gender			Reported Ethnicity				
Quartile	Apps Submitted	Number of Participants	Reported Avg. Age	Male	Female	NA	Black	Multi	Latino	White	NA
1	1-2 apps	68	11.90	22	25	1	39	7	0	1	1
2	3-4 apps	28	12.35	9	15	0	20	2	0	0	0
3	5-6 apps	15	13.10	5	8	0	8	5	0	0	0
4	7+ apps	9	12.80	4	5	0	5	3	1	0	0

The grouping of students into only four quartiles served as a practical means to evenly divide our participants into groups for comparison. Though the number of students that submitted apps for each group varies widely, the number of students that participated in submitting CT quizzes and surveys is more closely divided with the 1st and 2nd quartiles dropping off considerably and students in the 3rd and 4th quartiles contributing data more often (though the groups are still not equal in any area of data collection). In addition, breaking the students into 11 groups based on the exact number of apps submitted (with 11 being the highest number submitted by an individual participant) would be impractical for statistical analysis since the top three groups (with 9-11 apps submitted) would have a single participant in each, which would call into question the validity of the results as well as whether the results can be generalizable to a larger population.

This study is trying to establish if there is a causal relationship between participation in this after-school intervention and gains in understanding of computing concepts, belief in ability to use various 21st century skills, and change in interest and desire to persist in learning ICT/STEM. One part of determining if there is a causal relationship between an experimental condition and its effect is to reduce the likelihood that there are other explanations for the effect

(Shadish, Cook, & Campbell, 2002, p. 6). This study tries to control for validity errors by pulling students from 9 different locations, but having groups with only one member negates the possibility of pulling students from various locations and it also does not allow for any control of extraneous influences (such as student engagement in app building outside of the intervention).

Groups with only one member could affect the internal validity of the results. Internal validity refers to the ability to infer an effect caused by a treatment or the experience of a participant in a study (Creswell, 2005, pp. 290-291). Some threats to internal validity, as defined by Cook and Campbell (1979), include selection bias—where individuals more receptive to or familiar with a treatment are part of the experimental group—and history—where participants encounter information relative to the study outside of the intervention (Creswell, 2005, pp. 291-292; Morgan, Gliner, & Harmon, pp. 529-531). Having very small numbers of participants in a group does not provide any counterbalance for individuals that came in with coding experience or that gained additional experience from another source during the intervention. Therefore, placing students into fewer groups helped to increase the likelihood that a more diverse representation of students from different schools, of varying genders, and with different ICT/STEM backgrounds were able to filter through to the four quartiles and diminish the threats to validity. Participants from the 1st and 2nd quartile came from all nine schools while participants from the 3rd quartile came from seven schools, and participants from the 4th quartile came from three different schools (including a boy's school, a girl's school, and a mixed gender school whose participants were predominantly female).

Lastly, table 9 shows us the known ages, genders, and ethnicities of the students in each quartile. While not all participants provided demographic data, it is noteworthy that the majority of students that reported data were female, largely pre-teens, and predominantly black/African American. The intervention served members of underrepresented groups.

Outcomes Related to Computational Thinking

This section reports results related to research question 1: Was there a relationship between participant engagement and outcomes related to Computational Thinking? Data sources reported here will include number of apps completed, CT quiz results, and fieldnotes in order to help support the quantitative findings. Results from the Computational Thinking Quiz are reported first.

CT Quiz

The CT quiz was distributed online as a pre-test and a post-test exam each semester, but each individual student only had up to 3 opportunities to complete it as the spring pre-test was only available to new students. The first completed CT quiz counts as the participant's pre-test and the final completed CT quiz counts as a participant's post-test. Results in this section only include data from participants that completed at least 2 exams (*n* = 32). See Table 10.

Table 10

Participants Who Completed 2+ CT Quizzes

Quartile	Range of apps submitted	Number of participants
1	1-2 apps	9
2	3-4 apps	8
3	5-6 apps	9
4	7-11 apps	6

When comparing the quartiles for the pre-test, post-test, and change scores, students in quartiles 3 and 4 did better on the post-test on average than students in the lower quartiles. See Table 11. Furthermore, when considering the change score, the average scores for the 1st quartile did not change while the average for the 2nd, 3rd, and 4th quartiles rose by 1.04%, 14.82%, and

20.83% respectively. This shows that students who worked on more apps tended to improve from pre-test to post-test and do better on the post-test than those who completed fewer apps.

Table 11

Average CT Quiz Scores by Quartile

Quartile	Pre-Test Average	Post-Test Average	Change Score Average
1	23.15%	23.15%	0.00%
2	13.54%	14.58%	1.04%
3	28.70%	43.52%	14.82%
4	22.22%	43.06%	20.83%

CT quiz change scores were then analyzed by the grant team (Calandra, Renken, Cohen, Hicks, & Ketenci, under review). Findings indicated a correlation between the number of apps a participant worked on during the after-school intervention and their score on the CT Quiz. The authors also conducted a One-way Analysis of Variance (ANOVA) with CT change score as the dependent variable and quartile as the factor in order to discover whether there were differences in CT change scores across the groups. CT change scores were significantly related to number of apps completed in the 4 quartiles ($F(3, 28) = 3.62$, $p = .03$).

The researcher next took our analysis one step further by exploring the influence that completing relevant cookbooks may have had on the participants' CT Quiz performance. Relevant activities were those cookbook activities containing information that directly pertained to questions on the CT quiz. Table 12 shows how the CT concepts represented in the CT Quiz paired with those in relevant activities.

Table 12

CT Quiz Questions, CT Concepts, and the Relevant Cookbooks

CT Quiz Question	CT Concept	Relevant Cookbooks
1	Loop	6, 9, 10
2	Event (Sequence)	2, 8, 9, 10, 11
3	Variable	8, 9, 10, 11
4	Variable	8, 10, 11
5	Event (Sequence)	2, 8, 9, 10, 11
6	Event (Sequence)	2, 8, 9, 10, 11
7	Event (Parallelism)	2, 6, 8, 9, 10, 11
8	Event (Operation)	6, 8, 9, 10, 11
9	Event (Sequence)	2
10	Loop	8, 10, 11
11	Conditional	6, 9, 10
12	Conditional	6, 9, 10

For this analysis, participants were excluded if they did not work on at least one of the relevant cookbook activities, which were activities 2, 6, and 8-11. Students were again grouped into four new quartiles, with participants in the 1st quartile having submitted 1 of the relevant apps, the 2nd quartile having 2 relevant app submissions, the 3rd having 3, and the 4th having 4 or 5 submissions (see Table 13). No participant submitted all 6 relevant apps. Twenty-eight participant's data were used for this analysis. Correlations and analysis of variance were used to gauge the relationship between the number of relevant apps completed and the CT Quiz performance.

Table 13

Participant Groupings Based on the Number of Apps Completed with Avg. Scores

Quartile	Number of apps	Number of participants	Pre-Test Avg	Post-Test Avg	Change Score Avg
1	1	12	18.06%	19.44%	1.39%
2	2	8	25.00%	39.58%	14.59%
3	3	2	29.17%	37.50%	8.33%

4	4+	6	26.39%	47.27%	20.83%

A correlation was found between the number of relevant apps completed, post-test scores, and change scores. For the post-test scores, both correlation coefficients, Pearson's r and Spearman's r_s, found significance (Pearson's $r = .54$, $p = .003$; Spearman's $r_s = .60$, $p = .001$), but the change scores only showed significance using Pearson's r (Pearson's $r = .46$, $p = .014$; Spearman's $r_s = .36$, $p = .06$).

Using a One-way ANOVA, significance between the new quartiles and the CT quiz scores was only found in the post-test data ($F(3, 24) = 4.56$, F critical = 3.01, $p = .01$). Analysis with a Univariate General Linear Model produced the same result with an effect size ($\eta p^2 = .36$) of 36% and the observed power was .83, which shows that Type 1 errors are possible but unlikely (as there is approximately only a 17% chance of one occurring). Furthermore, Levene's test indicated equal variances between the quartiles in the post-test scores ($F(3, 24) = 2.93$, $p = .05$), so the Welch test was not necessary.

Post hoc analysis for the post-test scores found significant differences between the following quartiles:

- Quartiles 1 and 2 (Tukey: $p = .06$, LSD: $p = .01$, Bonferroni: $p = .09$)
- Quartiles 1 and 4 (Tukey: $p = .01$, LSD: $p = .003$, Bonferroni: $p = .02$).

Therefore, one might infer that the biggest difference was found between quartiles 1 and 4 using Tukey's HSD and the LSD exams, while quartiles 1 and 2 also showed a significant difference, but only when using the LSD test. Arguably, quartile 3 is problematic for this analysis since there were only 2 participants who completed 3 of the relevant apps. ANOVA results without this group still found significance in the post-test results ($F(2, 23) = 6.49$, F critical = 3.42, $p = .006$) and Levene's test indicating equal variances ($F(2, 23) = 3.33$, $p = .05$). Univariate GLM gave the effect size ($\eta p^2 = .36$) and power (.86). The Post Hoc tests again showed significance between quartiles 1 and 2 as well as quartiles 1 and 4 (with exception to the Bonferroni test between the first two quartiles):

- Quartiles 1 and 2 (Tukey: $p = .04$, LSD: $p = .02$, Bonferroni: $p = .05$)
- Quartiles 1 and 4 (Tukey: $p = .009$, LSD: $p = .003$, Bonferroni: $p = .01$).

Fieldnotes and Observation Data on CT

The following themes emerged from qualitative analysis and were used to answer research question 1: **student activity**, **instructional design, and social learning**. Student activity pertains to the apps that participants worked on, including customization and DIY projects. Instructional design relates to the elements of the intervention that encouraged productivity in app development, and which included scaffolding, tech support, and anything that helped students learn the curriculum content. Social learning revolves around the collaboration that occurred with mentors and facilitators or between participants in order to complete app projects. Observations from the field notes cannot measure the student's comprehension of Computational Thinking concepts, but the observations can reveal what the students accomplished during sessions of the intervention and what design elements assisted their learning.

Student Activity

Almost every student that attended the after-school intervention had a chance to start working on at least one app as we typically held off pre-testing until after a student's first week. This was done so that new students would be able to have an accurate impression of the intervention should they still be deciding between attending the intervention or another activity within the same after-school program. In most cases, students worked on cookbook 1 on their first period in the intervention, a project called Favorite Artist (an app themed around a popular African-

American artist). Ultimately, 120 students had apps that were reviewed. This was mainly due to a lack of student submissions rather than a lack of effort.

Among the students who did submit apps, not all projects were consistently uploaded, nor were the latest versions of their apps always uploaded. For instance, student 185 mentioned having a DIY app about a frog (he described it as being a redesigned version of the first cookbook project), but it was not visible on the App Inventor profile that he had open, so he may have used a personal account or deleted it. Similarly, student 114 added additional features to his trivia/quiz app project, which included adding a title screen. Unfortunately, a file with the changes to the quiz app was not submitted, so it is uncertain if he completed the coding. Lastly student 191 started building a DIY game that included two anime characters (a robot and a monster) and a detailed background, but it is unclear if he began coding the app. Nevertheless, he said that he found a video on making animations (presumably covering the code for cycling through images in a loop), and I recommended adding buttons for controlling their actions. This project was not submitted by the end of the spring semester. Student 191 also had plans for another game, having showed me a drawing and explained the concept of it, but he may not have gone beyond prototyping on paper. Ultimately, what these select examples illustrate is that more work was done during the intervention than can be recorded since not all work was submitted.

Most of the students in the 1st quartile worked on the first 2 apps, though several students worked on projects up to activity 8. In fact, 2 students worked on the Health DIY app though they did not include any code with their submission. Students that only submitted higher level activities (like student 648) may have completed earlier ones, but that cannot be determined. For instance, student 785 only submitted activity 1 though the researcher at his school confirmed that he was one of their highest achieving students. Unfortunately, all of his projects were removed from App Inventor, and we were never able to resolve why they disappeared from his account. We cannot confirm how many other apps he completed, so he has to be placed in the first quartile.

Students in the 2nd quartile mostly stuck to the first 5 activities, but some ventured further ahead to the advanced activities and even DIY activities (with only activities 8 and 9 not attempted at all). For instance, student 107 skipped further ahead to activity 11 in order to work on a game before backtracking to activity 5 in order to work at a more comfortable level. Student 715 showed the most ambition by creating a DIY app that was similar to the first 2 packets but on a much larger scale as her app crossed multiple screens and featured multiple musicians. Student 831 completed a DIY that saw her taking an existing app (the MLK app) and adding additional code so that it would read the inspirational quotes aloud. This DIY was a recommended challenge in the packet, but only a few students attempted it. One student started the Health DIY app, but the app lacked code.

The 3rd quartile had a broad range of interest in the activities with at least one attempt at each activity, including 1 Health DIY app and 3 other DIY activities. One DIY activity was the start of a game but it did not contain code, one added speech to the MLK app, and one performed a basic image swap when a button is pressed. Again, not all students worked in order as some skipped ahead to work on games (activities 8, 10, and 11). Lastly, a few students did not turn in every app that they were observed to have completed. For instance, student 545 worked on activity 10 for several periods but did not submit it and student 335 completed and even modified the appearance of activity 8 during the last period but did not have a chance to submit it, perhaps because he had additional features that he wanted to add.

Students in the 4th quartile had the deepest level of participation with multiple submissions for each activity. These students also completed the more DIY activities than the others, including 6 Health apps that contained some coding and 6 unique DIY activities. The most basic of these DIY activities re-themed the 1st app (done by students 114 and 573). Student 114 also started the group ninja app on his own (making artwork and setting up some of the components), but the submitted app lacked coding although he talked me through the steps necessary to get the image sprites to move around the screen on the final day. The top student, 507, created the other 3 DIY apps, with one being a remixed version of activity 11, having additional code and visual decorations and two being unique ideas: a fan app for a musician and a hypnotism app that contained flashing lights and a spinning pinwheel.

Instructional Design

Among the students that completed work, various aspects of the instructional design were successful at building interest in app building, with the app topics, ability to work out of order, and prospect of collaboration being particularly important for driving student activity. Many students liked the first two app activities since they revolved around a popular black/African American musician that many of the participants knew. The apps include short samples of two songs, and students tended to race to finish once they started hearing their peers testing out their apps. The 5th app activity, about Dr. Martin Luther King, proved to be very popular when students spoke about it in interviews, and many participants enjoyed finding quotes online while they worked on the app during the intervention. Activity 9 had students create a multiple-choice quiz/trivia game where they pick a topic and write questions and answers with feedback for the user. Participation was lower for this activity, but participants like student 114 based it around a topic that he enjoyed (Marvel comics) and had fun with it. The game apps (activities 8, 10, and

11) proved to be a catalyst for getting many students, male and female, to participate (like students 143, 153, 545, and 573) either due to the appeal of the resulting app or the fact that they were more challenging to complete. Students 143 and 153 worked on activity 11 together while students 545 and 573 gave each other support while working on activity 10. Several participants (like students 107 and 569) actually jumped ahead in order to work on games.

In addition to peer collaboration, scaffolding and support from mentors, facilitators, and peers. Most students required facilitator support getting started with creating accounts for the intervention's learning management system (LMS) or when logging into App Inventor for the first time. Beyond that, scaffolding was vital for some students to progress through the early cookbooks before they could become more independent, while others required support for idea generation, motivation to be productive, or just encouragement to keep trying.

Student 167 serves as a good example of how scaffolding helped someone become more comfortable with following the instructions and using the App Inventor platform. Student 167 needed a lot of support getting through the first two packets in particular, which took several periods for him to complete. It seemed as though he would only really work when the researcher was there to guide him, although he slowly became more independent as the academic year progressed. For example, by the time he reached activity 5, an app incorporating quotes from Dr. Martin Luther King, Jr., he was showing major strides in his ability to work independently. By the time he worked on the calculator project (activity 6), he only came to the undergraduate mentor and I with occasional questions. Student 167 did need some help getting started with cookbook 10 (a game app and one of the most difficult activities) since the activity included less direct instruction in the early steps, expecting students to rely on their previous experience to complete many of the initial steps. However, he was ultimately able to complete the activity largely on his own, refusing to go for an easier activity when offered.

Many of the participants socialized during class by either showing off their work to each other after finishing an app, asking each other or a mentor/facilitator for help, or, in a few rare occasions, collaborating via teamwork. This included teaming up to make one app on one computer or by working on the same cookbook simultaneously from separate computers while talking each other through the process. The girls at school 500 seemed to particularly benefit from a social environment. This was evident at school 500 via evidence in the data of their answering each other's questions, showing off completed work, soliciting feedback, and making the atmosphere more fun and engaging. Students 507, 544, 545, 562, 569, 573, and 591 helped each other the most, although there were some other pairings among other students as well. Student 507 was the highest achieving student in this group based on her number of apps completed and quiz scores, and it was noteworthy that she served as a resource to her peers by answering questions and making recommendations to her peers. Student 507 often sought to stay ahead of her peers, and only sought assistance from a mentor or facilitator. She also enjoyed showing off her projects to her peers after completion. One example of this in the data includes a demonstration of her DIY hypnotism app. In fact, student 591 then told her that the app's spinning wheel and flashing colors gave her a headache.

Collaboration also became an important motivator in one instance for students at school 100. While some of the students were working on cookbook activities during the spring semester, many others had lost interest in participating. The researcher divided these participants into two teams of 4 students. These teams were tasked by the researcher to try to recreate a ninja video game app. The teams were required to create their own characters, and animate these characters within the game (e.g., to move around the screen while executing an image swap based on which buttons are pressed by a user). One team consisted of several artists and they spent the rest of the semester engaged with working on character designs and storyboards for each character, so the researcher assisted them with the process of uploading these drawings and programming a simple image swap based on a button push. Most of this work was done by the group leader, student 177 towards the end of the semester.

Summary of Participant Outcomes

Ultimately, students in the 3rd and 4th quartile had the highest average post-test scores for the CT quiz, and the 4th quartile had the highest change in score from the pre-test to the post-test (nearly doubling from 22.22% to 43.06%). Furthermore, students with the most exposure to the cookbook activities that covered concepts that were most closely related to the contents of the CT quiz (cookbooks 2, 8-11) tended to perform the best on the post-test, with students that submitted 4 or more of the most relevant cookbooks having a post-test average of 47.27%, which was much higher than students that submitted 3 or less of the most relevant apps).

Scaffolding and collaborative learning was important for students in every quartile. Students in the higher quartiles tended to work on more of the DIY activities without guided instructions. Arguably, this shows greater confidence in their ability with coding and problem solving, but even a small percentage of students in quartile 2 were willing to experiment with DIY activities. Many of the students appreciated the freedom to skip around through the activities, but arguably some restraint may be necessary if students jump far ahead in order to create a game but do not have enough practice learning the platform as some students ended up going back to lower level activities in some instances.

Self-Efficacy Related to Participant's 21st Century Abilities

This section will report data related to research question 2: Was there a relationship between participant engagement (i.e., the number of apps completed) and participants' reported 21st Century abilities? Data sources for this section included the ICT-21Q along with a series of open-ended survey questions more specific to the intervention. Some interview and fieldnotes data will also be reported.

Similar to the CT quiz, students had up to 3 opportunities to work on the profile survey, but only the final survey is included as the post-test. The profile survey is much longer than the CT quiz, so it was more difficult for students to complete this task each time. Therefore, various parts of the profile survey will be included for a participant if they completed at least 50% of it, making it possible for participants to contribute data to some sections while being excluded from others. The profile survey consisted of multiple sections. For student data to be included in the ICT-21Q section, students must have completed a pre-test survey and a post-test survey at some point in the year (either over a single semester or the entire year). All student data from many of the other sections of the profile survey (like demographic data and short answer questions) could be included even if a student only completed either a pre-test or a post-test.

Profile Survey Data

The profile survey included the ICT-21Q and additional questions that reveal information about participants' belief in their ability to succeed and their background with coding and app building. The ICT-21Q remained consistent across the pre-test and post-test but additional survey questions varied across each survey. The pre-test included questions about the student's confidence upon entering the intervention and their previous knowledge. The post-test included short answer questions asking students why they joined the intervention and whether the students felt good about what they were doing in the intervention. The post-test also had questions about previous experience with coding and app building, but these questions were ignored since many students that answered the same questions in the pre-test changed their response to indicate having prior experience, which shows that they considered their time in the intervention as previous experience though these questions were for the timeframe prior to the intervention.

ICT-21Q

Participant pre-test, post-test, and change scores on the ICT-21Q were compared among the app completion quartiles used above. Table 14 notes the number of students for each quartile who completed the ICT-21Q pre- and post-tests as well as their score averages while Figure 7 shows the differences between the average scores on each survey.

Table 14

Number of Participants (n = 43) for the ICT-21Q and Avg. Scores for Quartiles

Quartile	ICT-21Q Participants	Pre	Post	Change
1	14	4.10	3.51	-0.59
2	12	3.72	3.71	-0.01
3	9	3.62	3.81	0.19
4	8	3.75	3.92	0.17

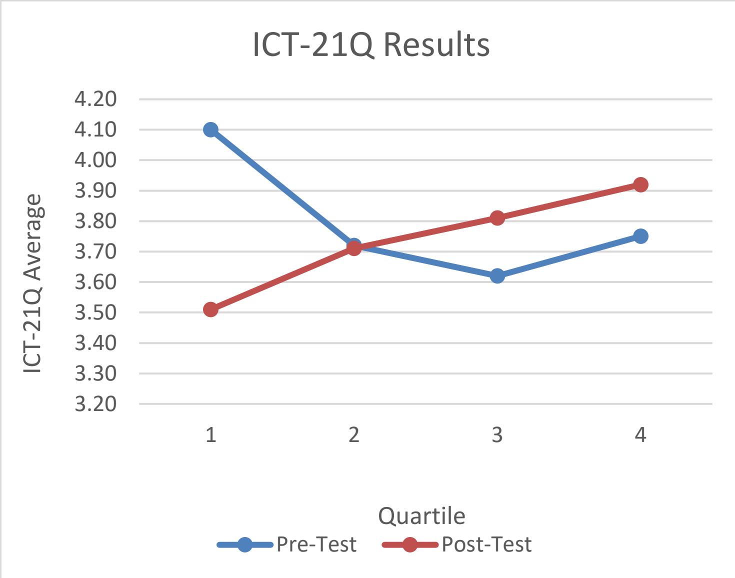

Figure 7. ICT-21Q averages for each quartile

For the pre-test results, students in quartile 1 had the highest average (4.10) while quartile 3 had the lowest average (3.62) and quartiles 2 and 4 were nearly equal (3.72 and 3.75). By the post-test results, the average score for quartiles 1 and 2 (3.51 and 3.71) decreased while the averages increased for quartiles 3 and 4 (3.81 and 3.92). On average, students in quartiles 3 and 4 were more likely to endorse their abilities to use 21st century skills while students in the lower quartiles were less likely to endorse their abilities.

ICT-21Q Correlation

SPSS was used to determine if there was a correlation between the number of apps each participant completed and the average scores for the ICT-21Q. Correlations were not statistically significant for the post-test (Pearson's $r = .19$; $p = .22$; Spearman's $r_s = .12$, $p = .46$) or the change score (Pearson's $r = .24$; $p = .12$; Spearman's $r_s = .23$, $p = .13$). Despite lack of statistical significance, the practical significance indicated in the direction of change scores reported in the previous section is worth noting.

Additional Survey Questions

The pre-test included the following questions to gauge the student's confidence and understand more about their background with app building and programming:

1. I feel confident in my ability to learn how to build phone apps (with responses falling on a Likert scale, 1-5, from "Strongly Disagree" to "Strongly Agree").
2. Have you built an app before (with responses standardized to "yes" or "no")?
3. Have you ever tried to write computer programs or code before?

The average pre-test responses for each quartile (given as a percentage or an average score) are presented in table 15.

Table 15

Miscellaneous Data from the Pre-Test (Fall and Spring) Profile Survey

Quartile	Participants	Question 1	Question 2	Question 3
1	28	4.04	50.00%	57.14%
2	17	4.24	35.29%	35.29%
3	8	3.50	12.50%	37.50%
4	7	4.43	71.43%	100%
All	60	4.07	43.33%	53.33%

No statistical significance was found for question 1 (Pearson's r = .08, p = .55; Spearman's r_s = .03, p = .83), 2 (Pearson's r = .01, p = .92; Spearman's r_s = -.11, p = .42), or 3 (Pearson's r = .08, p = .61; Spearman's r_s = -.06, p = .71) when the number of apps submitted was compared to responses to each of the pre-test survey questions. Interestingly, question 3 asks, "Have you ever tried to write computer programs, or coded, before?" From the four quartiles, 53.33% had previous programming or coding experience with 57.14% of respondents from quartile 1, 35.29% from quartile 2, 37.50% from quartile 3, and 100.00% from quartile 4 having experience. Unfortunately, it is not possible to determine the extent of the student's previous experience, which could have occurred once for fun or multiple times in a class or after-school program. Nevertheless, quartile 4 shows that coming into the program with experience did seem to help the students that persisted the longest though the exact nature of the previous experience is indeterminate.

Next, the post-test surveys included two short answer questions that will be discussed in this section:

1. Why did you choose to join the intervention?
2. Were there any times when you felt good at what you were doing?

Tables 16 and 17 show the most common responses from each quartile for the short answer questions and Table 18 shows what percentage of the responses qualify as a positive, negative, or neutral answer.

Table 16

Common Responses about the Participants' Motivations for Joining

Why did you choose to join the intervention?				
Quartile	Positive		Neutral	Negative
	Learn app building/fun	Peer recommendation	Just an option	Asked/made to go
1	6			2
2	5			
3	2	1	1	
4	1		1	

Table 17

Common Responses about the Participants' Sentiments towards the Curriculum

Were there any times when you felt good at what you were doing/yes?					
Quartile	Positive		Neutral	Negative	
	Learn app building /having fun	New experience / it was great / just being there	N/A	Did not like the intervention	Rewards /Coins
1	4	3		1	1
2	6	2	1		
3	5				
4	2	2			

Table 18

Percentage of the Responses as Positive, Negative, or Neutral Answers

Quartile	Participants	Score Ranking	Question 1	Question 2
1	32	Negative	6.25%	6.25%
		Neutral	0%	0%
		Positive	18.75%	21.88%
2	22	Negative	0%	0%
		Neutral	0%	4.55%
		Positive	22.73%	36.36%
3	12	Negative	0%	0%
		Neutral	8.33%	0%
		Positive	25.00%	41.67%
4	8	Negative	0%	0%
		Neutral	12.50%	0%
		Positive	12.50%	50.00%

Survey results revealed that students joined the intervention for a few different reasons, but they were mostly positive. Most participants (n =14; 32.43%) joined out of their own personal interest to build apps, a desire to try something new, or because friends recommended it. However, two students were compelled to join by their parents or teachers.

In order to answer the short answer question about whether they felt good about what they were doing, students first had to affirm in a separate question that there actually were times that they felt good (Q1, n = 22; Q2, n = 15; Q3, n = 8; Q4, n = 5) though not all students went on to respond to the short answer question. The responses to the short answer question, however, were slightly mixed. Most of the responders (n = 24) mentioned feeling good when they were making an app or just taking the time to try something new. One student (from Q1) felt good about knowing how to do something that others did not, and at least one participant (from Q4) mentioned feeling good since the work became easier for him or her to complete after each activity while another (from Q2) felt good about remembering how the blocks worked.

Lastly, though there may be a difference between the quartiles in terms of their average responses, the differences are not statistically significant. There was not a correlation between the student responses for question 1 (Pearson's $r = .04$, $p = .86$; Spearman's $r_s = -.04$, $p = .87$) or question 2 (Pearson's $r = .10$, $p = .24$; Spearman's $r_s = .02$, $p = .93$) and the number of apps built. Ultimately, these responses show that many participants were motivated to learn how to build apps, but some may not have been perhaps because their attendance was compelled by adults. Also, many students felt motivated upon completing activities successfully and some mentioned feeling good about the process as it became more manageable for them over time. This indicates that at least some students may have grown in their self-efficacy regarding their ability to use some 21st century skills (like ICT literacy and self-direction). Lastly, though there is not a statistically significant difference between the quartiles and whether they have good feelings about the intervention, each successive quartile gave more positive responses to the question. This may indicate that participants in the higher quartiles had more of an intrinsic satisfaction with developing apps.

Interview Data

Semi-structured interviews consisted of 14 questions overall. The following interview questions are related to research question 2:

- What made you want to sign up for the after-school intervention?
- Were there any times when you felt like you were good at what you were doing? If so, can you describe those times?
- How did making real apps that other people can download and use make you feel?
- Did you work with another student at any point?
- What were the most important or coolest things you learned from your time in the intervention?

Thirty-two participants completed an interview in the fall and 15 participants completed it in the spring. See Table 19 for a listing of interviewees by quartile.

Table 19

Number of Interview Participants per Quartile

Number of Participants	Fall	Spring
Quartile 1	10	1
Quartile 2	9	4
Quartile 3	4	7
Quartile 4	9	3

When asked why they joined the intervention, most interviewees were motivated to be in the program and learn app building (fall, *n* = 25; spring, *n* = 13) though a few may have been compelled by adults (fall, *n* = 3; spring, *n* = 1). Most students felt that learning how to build apps was the most important or coolest thing that they learned during the intervention (fall; *n* = 28; spring, n = 12). The students largely felt good about completing apps (fall, *n* = 26; spring, *n* = 11), which may have boosted their confidence during their time in the intervention. Some directly stated that they noticed their own improvement and felt more confident while working with App Inventor (fall, n = 5). Most students liked the idea of hosting their apps online and even making apps that could be beneficial for others (fall, *n* = 18; spring, *n* = 1). Lastly, collaboration was important for students who worked with mentors (fall, *n* = 12; spring, *n* = 12) or their peers (fall, *n* = 24; spring, *n* = 11). This may have played a role in helping the participants become more self-directed, particularly when considering the role of mentors and researchers in providing guidance while helping the students become more self-reliant. See Tables 20-24 for a listings of common student responses categorized by positive, neutral, and negative connotations.

Table 20

Common Interview Responses for Joining

What made you want to sign up for the after-school intervention?					
Quartile	Positive		Neutral		Negative
	Building apps, Similar interest (robotics)	Like STEM	Friends	Something New	No choice
Fall					
1	8	1			1
2	7		2		
3	3				2
4	7		1		
Spring					
1	1				
2	3			1	
3	6				1
4	3	1			

Table 21

Common Interview Responses for Things Learned

What were the most important or coolest things you learned from your time in the intervention?							
Quartile	Positive						Neutral
	Building /finishing app	Improving	Everything	Team work	Develop new skill	Working hard	NA
Fall							
1	9			1			
2	8						
3	4			1	1		
4	7			1			
Spring							
1	1						
2	3						1
3	6					1	
4	2					1	

Table 22

Common Interview Responses for Participant Feelings

Were there any times when you felt like you were good at what you were doing?								
Quartile	Positive			Neutral			Negative	
	Making /finishing apps	Becoming confident	MLK quotes	Getting help from friends	Rewards, coins	NA /Nothing	Jokes /socializing	Falling Behind
Fall								
1	6	1				2	1	
2	9	3	1					
3	3	1						1
4	8				1			
Spring								
1					1			
2	2				1	1		
3	6				1			
4	3							

Table 23

Common Interview Responses for Making Real Apps

How did making real apps that other people can download and use make you feel?					
Quartile	Positive		Neutral	Negative	
	Feels good about making apps	Wants to share them	Not sure	Waste of time, but practice	Afraid people won't like it
Fall					
1	5	7	1		
2	4	3	3		
3	2	2			1
4	2	6			
Spring					
1	1				
2	4	1			
3	5				
4	3				

Table 24

Common Interview Responses for Peer Collaboration

Did you work with another student at any point during the intervention?			
Quartile	Positive		Neutral
	Mentor	Other student	No collaboration
Fall			
1	5	8	
2	1	6	3
3	1	5	
4	5	5	
Spring			
1		1	
2	4	2	
3	6	5	
4	2	3	

A statistically significant correlation between the student responses and the number of apps built was not found for any of these questions. One finding that stands out comes with the question, “Did you work with another student at any point during the intervention?” Quartile 2 was the only group with members that reported that they did not collaborate, while it was important, to some extent, for the others.

Fieldnotes and Observation Data on Self-Efficacy

The themes of **affective domain and instructional design** that emerged from qualitative analysis were used to answer research question 2 in the following section. Affective domain pertains to the participants’ motivations, the personal relevance they see in the activities, the feelings of confidence they develop, and the growth they exhibit throughout their involvement in the intervention. The instructional design theme in part discusses the aspects of the intervention that may have helped students to become more confident. Observations from the field notes can reveal what affected student performance and led them to become more confident in their ability.

Affective Domain and Instructional Design

Some students may have demonstrated natural ability when it came to constructing apps based on the guided, cookbook packets by working alone with little help and finishing relatively quickly, but the students did not always express confidence in their ability to build the apps or code, particularly when it came to the DIY projects. As illustrated previously, student 167 was one student that was willing to try harder as the year progressed and he became more of an independent worker, but he did not necessarily express feelings of confidence about coding. For instance, he stated that coding was hard and he was not sure about his ability to get the DIY app started for his group project towards the end of the spring semester. He practiced by completing a game app (cookbook 10), but he did not work on assembling the DIY app with his group after making some character designs with the team.

Students 143 and 153 were bored of the cookbook material that they had been working on and were confident enough to skip far ahead in the activity packets and collaborate to work on activity 11 (which was the most advanced offering, a game called Space Invaders), but they confessed that they were just following the photos in the cookbooks and did not really understand the coding when I asked them how well they understood the programming. Later in the year, student 143 went on to attempt the Health DIY app (activity 7) and enjoyed working on it. He suggested during one interview that instead of giving students the cookbooks, facilitators should provide the participants with code and have the students figure out what to do with it. This arguably shows a gain in confidence for student 143, especially when previously he protested suggestions that I made for customizing his Space Invaders app under the assumption that he did not understand how to code earlier in the year.

Incidentally, some of the hardest working students (like 114, 131, 143, 191, 335, 377, 487, 507, and 785) tended to work alone or ask a facilitator for assistance, but not even all of these students worked on DIY projects. To counter the skepticism students felt in their abilities, GRAs and mentors tried to express that working on more of the packets helps them get used to the App Inventor environment and pick up new coding techniques. Furthermore, the key to making DIY apps is to combine what they learned in previous activities and apply it in new ways (i.e. remixing). We also let students know that they could consult us or do research on YouTube and coding forums to find new codes when they want to try something completely new. Still, many students felt reluctant to do DIY activities like the Health app (only 11 students submitted a health project and 7 submitted any other type of DIY activity), even among some of our higher achieving students like 377 and 167. Student 335 was in the 3rd quartile and he too was reluctant to work on the Health app, but he did submit one DIY project which involved adding additional code to his MLK app so the messages were read aloud. He also remixed his final app, activity 8, by adding additional visual design and making plans for adding sound effects, but unfortunately he did not have a chance to submit the app before leaving on the final day.

As part of the instructional design, scaffolding was one fairly important way for helping students go from being dependent to being more explorative later on in the year. For instance, student 715 submitted one of the most advanced DIY apps after working on it in the spring semester. During the fall, however, she needed heavy scaffolding to get used to using the system and to progress through some of the early packets (including testing and submitting the apps). Her DIY project was mainly a remix of cookbooks 1 and 2 in that it contained various media (pictures and MP3 samples) but it spanned across multiple screens with a navigation system connecting them all. I was not mentoring or collecting data at school 700 in the spring, so it might be the case that the undergraduate student mentor helped her figure out how to link he content across screens. Nevertheless, the expansive project pertained to a topic that would be of interest to her (it was a fan app for various musicians) and was of a size and complexity that not many others achieved.

Ultimately, not all students felt especially confident in their ability to code and work on DIY projects. Several students opted to skip the DIY projects and just stick to the guided activities. There were mini-DIY activities included in many of the cookbooks that were meant to get students to customize most of the apps, so more stringently pushing students to work on those might have made them feel better suited to build a project from scratch. On the other hand, some students really wanted to work on a DIY activity, but they may not have had enough tools to get through the coding. For instance, student 104 (an athlete) wanted to make a DIY football app comparable to Madden games seen on Xbox and other modern gaming consoles, and he would not compromise when I suggested ways that he could reasonably make a football game using techniques found in activity 10, Ladybug Chase. I suggested that the game would have a football player sprite run down a field while opponent sprites hone in on his location; sprites coming into contact before reaching the screen edge would stop the round. Student 104 did not like the idea as no one would want to buy it. The undergraduate mentor was able to work with the student a few days later as they made a more modest DIY app that remixed the first two apps and featured information about a football player instead (perhaps modeling as a digital sports card). The app was not submitted as the student may not have finished it before having to leave the intervention for sports practice, but this example illustrates the necessity of helping students work on a DIY project with a topic that they are passionate about while also keeping the complexity at a reasonable level of difficulty that they can master. Finding a proper balance between guided activity and DIY is an area that will need attention for future iterations of the intervention.

Summary of Self-Efficacy

The ICT-21Q did not show a statistically significant difference between the quartiles, but each succeeding quartile had a slighter higher average post-test score than the last. The pre-test to the profile survey further revealed that the 4th quartile had the highest level of confidence in their ability to learn how to build apps. The 4th quartile also had the most previous with app building and coding, which may play a factor in one's level of persistence and self-efficacy. However, the exact nature of their previous experience cannot be easily determined, as it could have been participation in a formal class or just a one-time, informal experience.

The survey further showed that the top two quartiles had the highest percentage of positive responses regarding whether the students felt good about what they were doing in the intervention. The most common responses among all students expressed positive feelings towards learning to build apps or just being exposed to a new experience. The positive feedback carried over to the interview as most students felt that the act of building apps was the most important thing that they learned in the intervention, and the students often felt good about finishing their apps. During the fall semester, students in quartiles 1 through 3 directly stated feeling more confident as the semester progressed. Unfortunately, confidence did not always carry over towards the completion of DIY activities, but students in the upper quartiles did engage with those activities more frequently.

ICT Affinity, Interest and Persistence

This section will report results related to research question 3: Was there a relationship between participant engagement (i.e., the number of apps completed) and their opinions about ICT subject matter and/or their desire to persist in ICT? Data sources reported here included the profile survey, interview data, and content from the fieldnotes that pertain to the participant's interest in ICT and interest in continuing to study the subject matter. The profile survey features the Technology and STEM Career sections of the STEM Semantic Survey along with miscellaneous questions regarding the participant's opinions, motivations, desire to persist in learning about app building, and background with coding. Similar to the CT exam, students had up to 3 opportunities to work on the profile survey. The profile survey was much longer than the CT quiz, so it was more difficult for students to complete this task each time. Therefore, various parts of the profile survey will be included for a participant if they completed at least 50% of it, making it possible for participants to contribute data to some sections while being excluded from others. For students to be included in the SSS portions, they must have completed a pre-test and post-test, but participant responses for the miscellaneous questions will be included even if only one survey is completed since they are inappropriate for a comparison from pre-test to post-test (for example, questions related to one's enjoyment of the after-school intervention or desire to continue building apps cannot be compared from pre-test to post-test). Interview data will be included from either semester and fieldnotes will incorporate any student with relevant data.

Profile Survey Data

The STEM Semantic Survey (SSS) questions focus on the topics of technology and STEM careers. See Table 25 for the number of students from each app completion quartile who completed each sections of the STEM Semantic Survey. All students did not necessarily complete every question on the profile survey, so it is possible for some students to have submitted a pre-test and a post-test without being able to participate in both parts of the SSS.

Table 25

Number of Participants that Completed a Pre- and Post-Test for the SSS

Quartile	SSS Tech	SSS STEM Career
1	11	12
2	11	12
3	8	9
4	7	7

For the STEM Semantic Survey (SSS), participants use a Likert scale to rank a series of descriptors for each of the subjects. Each scale was on a continuum (for instance, from boring to interesting), and participant responses were recoded to uniformly make a score of 1 correspond to a negative perspective and 7 to a highly positive perspective. Ultimately, the SSS is expected to reveal information about the participant's attitude towards ICT/STEM subject matter.

The SSS remained consistent across the pre-test and post-test, but there were additional profile survey questions regarding the students' opinions about ICT and desire to continue learning ICT/app building varied across each survey. The pre-test included a question about the students' feelings towards coding. The post-test repeated that question and asked additional questions about the intervention and their suggestions for improvement.

SSS: Technology

The results for the STEM Semantic Survey are less conclusive than the results for the ITC-21Q. For the questions about the student's opinion of technology, Quartiles 1 and 3 both decreased on average from the pre-test to the post-test while Quartile 2 increased slightly and Quartile 4 increased the most. While quartile 4 can conclusively be shown to have the most positive opinion regarding technology by the post-test, it is odd that quartile 3 would diminish by a relatively large factor while quartile 2 increased at all (see Table 26). So it cannot be said that students progressively have a higher opinion about technology with each successive quartile, but all of the students had a mostly positive attitude towards technology (with average post-test scores ranging from 5.38 to 6.29 out of 7 possible points).

Table 26

Averages for the Technology Portion of the STEM Semantic Survey

Quartile	Pre	Post	Change
1	5.80	5.45	-0.35
2	6.07	6.13	0.06
3	6.09	5.38	-0.71
4	5.97	6.29	0.31

While calculating whether there is a correlation between the numbers of apps each participant completed and the average scores for the SSS technology questions, the results were not statistically significant for the post-test (Pearson's r = .15; p = .37; Spearman's r_s = .09; p = .59) or change score (Pearson's r = .10; p = .55; Spearman's r_s = .10; p = .56).

SSS: STEM Career

Correlation analysis for the post-test (Pearson's r = .18, p = .28; Spearman's r_s = .12, p = .45) and change scores (Pearson's r = -.11, p = .51; Spearman's r_s = -.10, p = .56) showed that the amount of work done in the after-school program did not significantly correlate with the student's interest in STEM careers. Despite the lack of statistical significance, all students in the first three quartiles slightly increased in their interest in pursuing a career in a STEM field while the 4th quartile maintained the highest average. Notably, the largest change was for students who completed the fewest apps (see Table 27).

Table 27

Averages for the STEM Career Portion of the STEM Semantic Survey

Quartiles	Pre	Post	Change
1	4.73	5.43	0.71
2	5.73	5.93	0.20
3	5.77	5.82	0.06
4	6.09	6.08	-0.01

Additional Survey Questions

In addition to the SSS, the pre-test and post-test of the survey asked students for their opinion about coding:

Which sentence best describes how you feel about coding?

a. I code often and am usually successful.

b. I've tried before, but I don't feel very good at it.

c. It was a one-time thing.

d. None of these answer choices really apply.

e. I'm still not sure.

The average responses for each quartile are presented in table 28.

Table 28

Comparison of Responses Regarding Students' Feelings about Coding

Pre-Test Answers	Post-Test Answers	Responses
10	8	I code often and am usually successful.
4	4	I've tried before, but I don't feel very good at it.
2	5	It was a one-time thing.
2	1	None of these answer choices really apply.
2	2	I'm still not sure.

Isolating the survey results to only the twenty participants that answered the question on both the pre-test and post-test, a very meaningful analysis is difficult to make. Seven students did not change their answer, two students (one in quartile 1 and one in quartile 3) went from claiming that they code often and successfully to claiming that they have tried it before but were not successful at it, and two students in quartile 4 changed their answers in the opposite way, claiming to code often and successfully. All of the other answer changes were using the more ambiguous responses at one point: "it was a one-time thing," "none of these answer choices really apply," and "I'm still not sure." Arguably, the group as a whole just feels more reserved in the post-test than the pre-test given the decline of responses claiming to be successful and the increase in responses just saying that coding was a "one-time thing."

In addition to the participant's feelings about coding, the profile survey's post-test also included the following multiple-choice questions:

1. How much did you like what you were doing (Likert Scale, 1-4, low to high)?
2. Would you like to continue with activities like this intervention?
3. Will you create apps on your own outside of the after-school program?

The following short answer questions were also included (some questions are rephrased):

4. What was your favorite part of the intervention?
5. What was your least favorite part of the intervention?
6. What would make these activities more fun?

Responses to questions 2-3 were categorized as being positive, neutral, or negative based on the participants giving an affirmative, indecisive, or negative answer to the question. Responses to questions 4-6 were categorized as positive, neutral, or negative based on the author's analysis of the statements that the participants made. Table 29 gives either the averages for the responses or the percentages of responses that fall under each category. Tables 30-32 give the most common responses from each quartile for the short answer questions.

Table 29

Additional Data from the Post-Test (Fall or Spring) Profile Survey

Quartile	Participants	Question 1	Score Ranking	Question 2	Question 3	Question 4	Question 5	Question 6
1	32	2.89	Negative	34.38%	28.13%	12.50%	65.63%	9.38%
			Neutral	0%	40.63%	18.75%	12.50%	43.75%
			Positive	62.50%	28.13%	65.63%	18.75%	28.13%
2	22	3.29	Negative	22.73%	9.09%	0%	50.00%	4.55%
			Neutral	0%	54.55%	18.18%	9.09%	31.82%
			Positive	63.64%	31.82%	68.18%	22.73%	22.73%
3	12	3.45	Negative	8.33%	8.33%	0%	50.00%	33.33%
			Neutral	0%	50.00%	8.33%	0%	58.33%
			Positive	83.33%	33.33%	83.33%	41.67%	8.33%
4	8	3.43	Negative	0%	12.50%	0%	50.00%	0%
			Neutral	0%	25.00%	0%	0%	62.50%
			Positive	75.00%	37.50%	75.00%	12.50%	12.50%

Table 30

Common Short Answer Responses about Favorite Parts of the Intervention

What was your favorite part of the intervention?									
Quartile	Positive		Neutral					Negative	
	Making Apps/ Everything	Working with computers	Coins	Art/ Creative	Collaboration	Don't know/NA	Fun	Nothing	Surveys
1	20	1	1			4	1	1	3
2	14	1		2	1	1			
3	10					1			
4	6								

Table 31

Common Short Answer Responses about Least Favorite Parts of the Intervention

What was your least favorite part of the intervention?												
Quartile	Positive		Neutral	Negative								
	Nothing	Leaving	Don't know	Code /App building	Difficulty	Everything	App quality	Surveys	Timing /length	Conflict with others	Rewards	Other
1	5	1	4	7	5	1	2	3	2	1		2
2	5		2	4		3	1		1			
3	5				2			2		1	1	
4	1				1		2			1		

Table 32

Common Short Answer Responses about Improving Activities

What would make these activities more fun?										
Quartile	Positive Neutral			Neutral			Negative			
	Nothing	More classes	More peers	NA	More apps/ games	Collab-oration	Do other /fun stuff	Less surveys	More reward	Less conflict
1	6	1	2	1	12	1	2	1		
2	5			2	4	1			1	
3	1			0	5	2			3	1
4	1				4	1				

Question 4 (what was your favorite part of the intervention?), for instance, had positive responses that revolved around making apps, working with computers, or everything about the intervention being their favorite part. Neutral responses were statements that did not explicitly mention app building but were not critical of the intervention. Negative responses did not credit any aspect of the intervention or they referenced the surveys rather than app building. On the question of one’s least favorite part of the intervention, more students in the lower quartiles had complaints about the process of app building (like coding, typing, and reading instructions). Some of the other least favorite aspects of the intervention included the quality of the apps they created, the amount of time it took to make one app, the lack or quality of rewards and various difficulties that were experienced (including trouble testing or uploading the apps, difficulty re-membering the URL for the LMS, or just needing to ask for help).

Conversely, the question soliciting suggestions for improving the intervention had more positive responses in the lower quartiles and quartile 3 had the most relatively negative demands. Students in quartiles 1 and 2 more often felt that nothing could be done to improve the intervention or they recommended adding more sessions and bringing in more of their peers. Neutral responses included adding more apps or games and allowing for collaboration. These are neutral responses since “adding more apps” could simply imply that more activities is beneficial or it could imply that the current activities were not that interesting and could be replaced. Collaboration is neutral since we wanted students to collaborate but some teachers discouraged it due to the noise that comes with it. So, unfortunately, some students may not have been aware that we were in fact happy for them to team up with their peers. The negative responses included allowing students to do other things (besides app building), having fewer surveys, offering more rewards for activity completion (due to students wanting extrinsic motivation), and removing sources of conflict (one of the school teachers was described as being “bossy”).

A correlation exists for question 1 (Pearson’s r = .27, p = .024; Spearman’s r_s = .35, p = .003) and question 4 (Pearson’s r = .27, p = .026; Spearman’s r_s = .26, p = .033). For question 4, each successive quartile gave a higher percentage of positive responses with the top quartiles tending to have few neutral and no negative responses. Levene’s test found equal variance in question 1 (F(3, 66) = .513, p = .68) but not question 4 (F(3,63) = 11.135, p < .001). Despite the correlation, no significance was found using one-way ANOVA for question 1 (F(3, 66) = 1.97, F critical = 2.74, p = .13). Univariate GLM gave an effect size (ηp^2 = .08) and observed power (.48). For question 4, the Welch test is not possible since at least one group has 0 variance. Nevertheless, this data shows that Quartiles 3 and 4 liked the intervention more than the lower quartiles as their averages were higher for question 1 (how much did you like the program?) and their answers for the other questions had more positive responses on average for questions 2 (would you like to continue with activities like this intervention), 3 (will you create apps on your own), and 4 (what was your favorite part of the intervention).

Interview Data

The interview consisted of 14 questions with 36 participants completing the interview in the fall and 16 completing it in the spring. Table 33 shows the number of participants that contributed data from each quartile.

Table 33

Number of Interview Participants per Quartile

Number of Participants	Fall	Spring
Quartile 1	10	1
Quartile 2	9	4
Quartile 3	4	7
Quartile 4	9	3

The following interview questions addressed the participant's interest in ICT content and the intervention curriculum, their desire to persist in learning about ICT content, and the factors that prevented access to the intervention:

1. How much did you like what you were doing?
2. What was your favorite part?
3. What was your least favorite part?
4. What kind of things affected your ability to come to the intervention?
5. What would help you or make you want to come to the intervention more often?
6. Would you like to continue with activities like these?
7. Will you create apps on your own outside of the program? Why or why not?
8. How would you make these activities more fun?

Student replies were indexed in categories based on whether they were positive, neutral, or negative responses to the given interview question. Tables 34-41 contain common responses for the eight most relevant questions.

Table 34

Common Interview Responses about Interest in the Intervention

How much did you like what you were doing?			
Quartile	Positive	Neutral	Negative
	Liked the intervention	Kind of liked it	Didn't like the intervention
Fall			
1	7	2	1
2	8		1
3	4		1
4	7	1	
Spring			
1	1		
2	4		
3	6	1	
4	3		

Table 35

Common Interview Responses about Participant's Favorite Part of the Intervention

What was your favorite part of the intervention?						
Quartile	Positive			Neutral		Negative
	Building apps	Gaining confidence	Sharing apps /Helping others with an app	Peers /Mentors	The song from the first cookbook	Rewards
Fall						
1	8					2
2	8				1	
3	3			1		1
4	7		1	1		
Spring						
1	1					
2	3			1		
3	5	1		1		
4	3					

Table 36

Common Interview Responses about Participant's Least Favorite Part of the Intervention

What was your least favorite part of the intervention?										
Quartile	Positive	Neutral	Negative							
	Nothing	Not sure	Tech issues	Too easy	Teacher	No breaks	Surveys /Math	Work alone	Peers Noise	Missing other programs
Fall										
1	4		4		1	1				
2	3		2	1	1		2			
3	1		3				1			
4			3					1	2	
Spring										
1							1			
2		1					3			
3			1				2		1	1
4	2		1							

Table 37

Common Interview Responses about Attrition

What kind of things affected your ability to attend the intervention?					
Quartile	Neutral	Negative			
	Nothing/not sure	Sports, band, home/family, nutrition, etc.	Tutoring	Conflict with teacher	Nervous of the program
Fall					
1	2	3	1	1	1
2	1	5	2		1
3	3	2	1		
4	4	4	2		
Spring					
1		1			
2	2	2			
3	2	3	1		
4		3			

Table 38

Common Interview Responses about Persistence in Learning

Would you like to continue with activities like these?			
Quartile	Positive	Neutral	Negative
	Yes	Maybe	No
Fall			
1	6	2	
2	7	1	
3	2		1
4	7		
Spring			
1	1		
2	2	2	
3	6	1	
4	3		

Table 39

Common Interview Responses about Persistence in App Building

Will you create apps on your own outside of the after-school program?			
Quartile	Positive	Neutral	Negative
	Yes (home or elsewhere)	Maybe	No
Fall			
1	5	1	1
2	5	2	1
3	2		1
4	2	2	1
Spring			
1	1		
2	3		
3	4	1	2
4	3		

Table 40

Common Interview Responses about Improving the Intervention

What would help you or make you want to come to the intervention more often?							
Quartile	Positive	Neutral			Negative		
	More apps	Competition	Teamwork	Instagram	Remove distracting students	Food	Better apps
Fall							
1				1			
2							
3							
4							
Spring							
1							
2	1		1				
3					1	3	1
4		1				1	

Table 41

Common Interview Responses about Improving the Activities

How would you make these activities more fun?												
Quartile	Positive		Neutral					Negative				
	Bigger class	Good as is	Share apps	Group work	More coding and games	Add own inter-ests	Not sure	New cook-books	Free time	Motivate students more	Prizes	No surveys
Fall												
1		2	2		5	1			1			
2	1	1			2	1		1	2		1	
3				1	2					1	2	
4				3	1	4					1	
Spring												
1												1
2	1				1				1		2	
3				2	1		1	1			1	
4				2	1						1	

When asked how much they liked what they were doing in the intervention, students in the upper quartiles predominantly gave positive comments in the fall (Quartiles 3 and 4, n = 11). The lower quartiles gave more neutral or negative responses (Q1 and Q2, n = 3) though even the 3rd quartile had 1 negative (the student did not enjoy the intervention) and quartile 4 had 1 neutral comment (the student kind of liked it). In the spring, however, the responses were almost entirely positive though one student in the 3rd quartile gave a neutral response (which she increased from the more negative comment the semester before). When asked about their favorite part of the program, most students mentioned app building (fall, n = 26; spring, n = 10) and one student in the spring (Q3, n = 1) mentioned that gaining confidence was his or her favorite part. Some students enjoyed working with friends in the fall (Q3 and Q4, n = 2) and either friends or mentors in the spring (Q2 and Q3, n = 2). Lastly, some students cited rewards (Q1, n = 2; Q3, n = 1) as their favorite part.

When asked about their least favorite part, the results were more diverse. Many students across the quartiles were positive and stated that they did not have a least favorite part (fall, n = 8; spring, n = 2). However, the biggest source of frustration stemmed from problems with technology, like App Inventor stalling when testing (fall, n = 12; spring, n = 2). Other frustrations included conflict with teachers (n = 1), peers creating a noisy environment (fall, n = 2; spring, n = 1), and the need to take so many surveys (fall, n = 2; spring, n = 6). It should be noted that tech failures were not cited as causes for attrition. The most common causes of attrition included attending tutorial (fall, n = 6; spring, n = 1) and participating in other activities at school—including sports and band—or home—including family activities/chores (fall, n = 14; spring, n = 8). Interestingly, two students in the fall cited their initial nervousness of being in the program as a barrier at first (Q1, n = 1; Q2, n = 1). Quartile 4 was the only group that left positive replies (indicating that they did not have a least favorite part of the intervention), and they had the lowest percentage of negative responses (indicating some frustration with technical difficulties).

When asked if they would like to continue in after-school programs similar to this intervention, 34 students expressed that they would like to continue in similar programs (fall, n = 22; spring, n = 12) though 1 student in the fall directly stated that she does not want to continue (Q3, n = 1) though that response became a more neutral “maybe” in the spring. When asked if they will continue to build apps from home or elsewhere, 25 students (fall, n = 14; spring, n = 11) said that they would consider doing so. However, 6 students in the fall (fall, n = 4 evenly distributed; spring, n = 2 from Q3) said that they were not interested in continuing to build apps.

When asked for ways that would make them want to come to the after-school intervention more often, one student in the fall (Q1, n = 1) recommended adding an app that would be like Instagram, a photo sharing site. Students in the spring more generally recommended adding more app activities (n = 2), allowing teamwork (n = 1), getting rid of distracting students (n = 1), adding competition (n = 1), and giving them food (n = 4). In the context of this question, negative replies would include requests for rewards (since they are not motivated exclusively by their work but also by a prize), removal of distracting students (this points out that peers could be disrupting the learning experience of others), and a request for better apps (which shows that there may be dissatisfaction with our course offerings). Quartile 3 stood out with this question as the group saw 71.43% of their respondents give negative replies in the spring.

A similar question asked students for ways to make the activities more fun, and the responses were mostly the same. For instance, students recommended more coding, apps, or games (fall, n = 10; spring, n = 3), larger class sizes (fall n = 1; spring, n = 1), allowing group work and even collaboration with coding professionals (fall, n = 4; spring, n = 2), DIY projects (fall, n = 6), helping them share apps online (fall, n = 2), and competition (spring, n = 2). While some students requested diversions, like rewards as a source of extrinsic motivation (fall, n = 4; spring, n = 4), some students made direct recommendations for the instructional design, such as requesting improvements to the cookbooks (fall, n = 1; spring, n = 1) or more ways to motivate them to work (fall, $n = 1$). In the fall semester, quartile 3 stood out in that they had more negative feedback in that they ($n = 3$) needed more motivation and wanted more prizes. The participants also wanted more group work.

For statistical analysis of the interview data, positive responses were given 1 point, neutral responses received 0 points, and negative responses received -1 points. Skipped questions were not given a score. Students that gave multiple responses to a question either had their response scores averaged (for instance, a student that responded positively and negatively to a single question would be awarded a score of 0) or weighted more towards the reply that received more consideration from the student (for instance, a student may have given two answers but went more in-depth on the positive response rather than the neutral response). In the fall and spring semester, a correlation was not found between the interview responses and the number of apps that students submitted.

Fieldnotes and Observation Data on Interest and Persistence

The themes of **affective domain, barriers and conflicts, and socio-cultural norms** from the researchers' fieldnotes were used to answer research question 3 in the following section. Affective domain pertains to the participants' affinity for coding and the personal relevance they see in the app building activities. Barriers and conflicts incorporates the known causes for attrition, student complaints, and sources of conflict that take students out of the learning experience or cause them to lose interest. Socio-cultural norms involve any content pertaining to the cultural interests of the students, such as an interest in creating an apt that would benefit their community or family. Observations from the field notes can reveal what may have caused students to build or lose interest in app building/ICT and what affected their desire to continue learning ICT.

Affective Domain

Some students had interests in line with building apps. For instance, student 114—one of the top students of the intervention that submitted 10 apps and contributed to most of our data collection—was naturally interested in ICT/CS subject matter. For instance, during the Christmas break, he took time to build an app using a different VPL platform. At the end of the year, he asked for recommendations of computer languages that he should go on to learn. Another top student, 507 (with 11 apps, including 2 that were completely original), similarly mentioned that she developed an interest in learning a more conventional coding language though she did not necessarily want a career in computer science. Another student, 177, also had motivations for attending the intervention. He stated that he wanted to become a game developer and that he wanted to make an app like the popular video game Super Smash Brothers. He was in high school and was taking engineering class where he learns some code and he had previous experience with app building by trying to build one app on Scratch. Student 131 too noted that the curriculum was helping him with a project for a class, but he still had some uncertainty about his ability with coding. Many of these students in the upper quartiles had a natural interest in coding/programming, although at least one (student 507) may have gained interest as a result of being in the program.

Some students had interests that were only tangentially related in app building. For instance, several students (107, 146, 342, and 114) were interested in making digital art though only one was specifically interested in studying computer science. Nevertheless, many students were interested in learning how to build apps for smartphones, and several asked if they could put their apps on Google Play or the App Store towards the start of the intervention.

Unfortunately, not all students were interested in being part of the app building intervention, which suggests that they were not given a choice to be there or not. These students tended to socialize, play video games, or surf the web fairly often during the intervention, even on their first day in some cases though others gave the program a chance for several periods before giving into distractions. Many students attended the intervention for some period of time but did not want to give assent for participating in this study, with some actually participating and others not making as much of an effort before leaving.

Conversely, some students worked diligently but were frustrated by a teacher preventing their socializing, and so they left the intervention (such was the case for students 342 and 323). Lastly, some students that were introduced to the intervention while being part of a control group expressed interest in joining and many returned for a period or two but could not commit due to their participation in sports or other programs. It is possible that advertisement for the intervention was not as prolonged or widespread, which may have led teachers to recruiting students that may not have attended otherwise. Though some students were forced into coming to the intervention, many gave it a chance and grew to like it. For instance, at school 500, many of the students that attended originally ended up leaving after the first week when band started so a large group of girls were recruited to join the program. They were reluctant to participate on the first day until one of their classmates finished the first app and successfully tested it, which spurred the new students to work harder to figure it out. The students grew to like the program and left predominantly positive feedback in the interviews. One of these students ended up becoming our top app creator (student 507 with 11 apps, including 2 completely original, DIY apps), and she often stayed behind late to complete her work. During the spring semester, some of the students returned of their own choice, but some were forced to rejoin in the while others attended different programs. This caused further friction at first with the school's teacher/moderator, but the students eventually became more willing to participate through the encouragement of the researcher and undergraduate mentor. Students 342 and 323 at another school similarly told a researcher that they were not willing to rejoin with their school's instructor serving as a moderator. These students at both schools did state that they did not mind the intervention, mentors, or researchers, but the faculty drained their interest.

Regardless of the reason why many entered the intervention, many students were happy to finish their first apps and developed an appreciation for the coursework. Student 383 mentioned that she was interested in coding and did not realize it was something that she could do so easily. Student 507 asked if we could branch out and program a robot, but we had to explain that her after-school program might have a different robotics intervention. Ultimately, student 591 stated that though they were forced to come, they ended up liking the intervention and most of her peers said that it was fun. Even in the spring, two students that were reluctant to rejoin (545 and 573) ended up having fun when I challenged them to skip ahead and complete one of the most difficult apps, a video game, and just try to rely on each other for support.

Barriers and Conflict

Difficulties with technology served as an impediment for students and may have spurred attrition though no one directly stated it. At the most extreme, some technical issues with App Inventor involved students losing files from their account. Student 785, who up to that point was the furthest along at his school and often aided his peers, did not have any of his apps on his student account towards the end of the fall semester. He did not have another Google account (or we checked his other account and the files were not there either). I conversed with someone that works for App Inventor/MIT about recovering the apps, but I never heard back from the MIT developer after our initial conversation and neither I nor another researcher ever found out if the student had his apps restored on his account since it was the end of the semester. Unfortunately, none of the apps were submitted to the LMS previously. Student 785 did not return in the spring, but there may have been additional reasons for his attrition.

Other technical issues with App Inventor included that it would stall when students were attempting to test their apps and that led the intervention team to bring laptops to the school so that students could open their projects at a testing station since they loaded more reliably on the laptops. In addition, some apps would not work properly though the blocks were coded correctly; the key to dealing with this issue was usually to disconnect the problematic chain of blocks and reconnect them (sometimes needing to delete the leading Event block and replace it with an identical one). Students also tended to forget to upload their completed apps to the LMS. We made a separate handout available with the steps for uploading projects, but a stricter approach to uploading might have been necessary (such as barring access to the next file without showing the program directors a receipt for submitting a previous project

To a lesser extreme, most participants had trouble logging into the intervention LMS and App Inventor website, and this was usually resolved by resetting passwords to the LMS, increasing the amount of permitted login failures, and getting students to carefully type in their student email address in App Inventor. Some login problems were more serious, with student 333 not having a Google account with the school at all, so he had to use my account and could only work on days when I was attending his school. Student 177 was not able to log into App Inventor on a school desktop computer and had to use one of the intervention team's laptops (this issue was never resolved and it remains unclear how the problem started). He rarely worked after the mid-point of the fall semester, and not having easy access to his account might have contributed to his loss of interest. Other students experienced more temporary blocks to App Inventor, like students 168 and 185. There were also issues with students using the intervention's LMS as participants often forgot their password though we asked them to setup their accounts with their student email and lunch number as their usernames and passwords. Thus, students had trouble with technology at various points, technical issues were not directly cited by them as a cause for attrition, though, on the other hand, outside activities were.

Sports and outside activities were the major known reason for attrition. Many students played basketball, golf, track, football, dance, and band practice. There were academic reasons for missing class at times as well, including tutoring, participation in a quiz bowl competition, and a field trip to Washington, DC. Lastly, some students left the entire after-school program, moved locations from the school, or had duties at home that came first.

Ultimately, every student had different motivations for being part of the intervention. The two top content producers, ironically, had opposite motivations going into the program with one being naturally interested in ICT subject matter (student 114) and the other being placed into the intervention by others, but growing to appreciate it (student 507). While program content seemed to have captured student interest, having a social environment was fairly important for many students. This social environment included peer collaboration and working with mentors.

Summary of Interest and Persistence in ICT

The STEM Semantic Survey portion of the profile survey revealed that students in the 4th quartile were the most interested in technology and careers in STEM fields, but the other quartiles' averages fluctuated so that there was not a significant difference between the groupings. Among the additional profile survey questions, students were asked to rank how much they liked what they were doing on a Likert scale, and the 3rd and 4th quartiles had the highest averages though there was not a significant difference between the quartiles. On the other hand, there was a correlation between the number of apps completed and one's ranking on the Likert scale.

The participants' favorite part of the intervention was app building among all the quartiles according to the profile survey and interviews. The profile survey reported that the least favorite part of the intervention was coding among a sizable number of participants in the 1st and 2nd quartiles while the interview cited technical difficulties (like trouble testing apps) as the biggest problem.

When asked if they would like to continue in similar activities, the 3rd quartile gave the most positive responses (followed by the 4th). When asked if they would like to make apps on their own outside of the intervention, the 4th quartile had the most positive responses (followed by the 3rd). This shows the upper quartiles were the most likely to persist in engaging in ICT and related activities. The interviews actually showed the quartile being the most likely to continue in the fall while the 1st and 4th were equally interested in continuing in the spring. The 2nd quartile was the most likely to keep making apps in the fall while the 1st and 4th were the most likely in the spring. The 4th were still likely to persist, but the 1st quartile also showed some interest in the interviews (though only 1 participant was responding for this quartile in the spring).

Unfortunately, all students did not join the intervention of their own free will, but several that were placed in the intervention grew to like it while a few students cited conflict with their teacher (from the school, not the researchers) as being their barrier towards persistence. Outside activities were also a barrier for students (including sports, nutrition, tutoring, and duties at home). By and large, though, students were interested in app building, and a few asked if they could host their work in an app store.

5 DISCUSSION

This chapter consolidates the findings from the research, compares our results to those of other after-school ITC/STEM research projects, and makes suggestions to help guide future research. Differences between the quartiles are emphasized when discussing outcomes related to computational thinking, self-efficacy, participant opinions about ICT and the intervention, and the student's desire to continue learning app building and coding. The limitations of this study are also addressed in this chapter.

Quartile Compairison of Engagement and Outcomes Related to Computational Thinking

Participant Engagement

Participant engagement was determined solely by the number of apps submitted by the participants. The apps that researchers have on hand are the only consistent way to determine the amount of work that students have done during the intervention as not all sites have alternate records of the students' daily activities. Researcher fieldnotes do provide information about the student's daily work, but researchers a) were not able to closely monitor each student to know exactly how much work was completed for every project that students attempted, b) did not attend every meeting to see exactly what students were doing (researchers were typically assigned to multiple sites each week and had to split their time between them), and c) did not all write fieldnotes (only two researchers created fieldnotes during the year). Therefore, the apps that students submitted provide the best tangible evidence across each school site of the student's level of engagement.

Furthermore, attendance records cannot be considered as a measure of engagement for various reasons. First, the teachers assigned to the intervention from each school were in charge of recording attendance, but their records were not always completed, contained entries for students that left early, and left off students that came in late (but were able to participate during most of the session). In addition, not all students actively participated while attending the intervention. If some students opted to work on other activities (like homework for classes or just socializing with their friends), then attendance records alone would not account for their behavior during the intervention.

App Completion

The main focus of the after-school intervention was to get students to become interested in app building specifically and ICT more generally. Table 42 highlights the amount of apps submitted by the students in each of the quartiles, including the number of students that submitted DIY projects and whether the DIY projects incorporated some coding (complete or not) or only focused on layout and design. As the table showcases, the higher quartiles completed more apps per student and more students in the upper quartiles attempted to make DIY projects, which included remixes of old projects and completely original ideas. Figures 8 and 9 further illustrates the differences between the quartiles in their completion of our offered app activities. The charts show that the two lower quartiles tended to stay within the first six activities while the higher quartiles had more diverse experiences, which in turn gave them more exposure to different computational thinking concepts on which they were tested through our CT quiz.

Table 42

Student App and DIY Submission Numbers

Quartile	Total Students	Total Apps	Students w/DIY Apps	Total DIY Apps	DIY apps with code	DIY apps un-coded
1	68	99	2	2	0	2
2	28	99	3	3	2	1
3	15	82	4	4	3	1
4	9	75	8	13	12	1

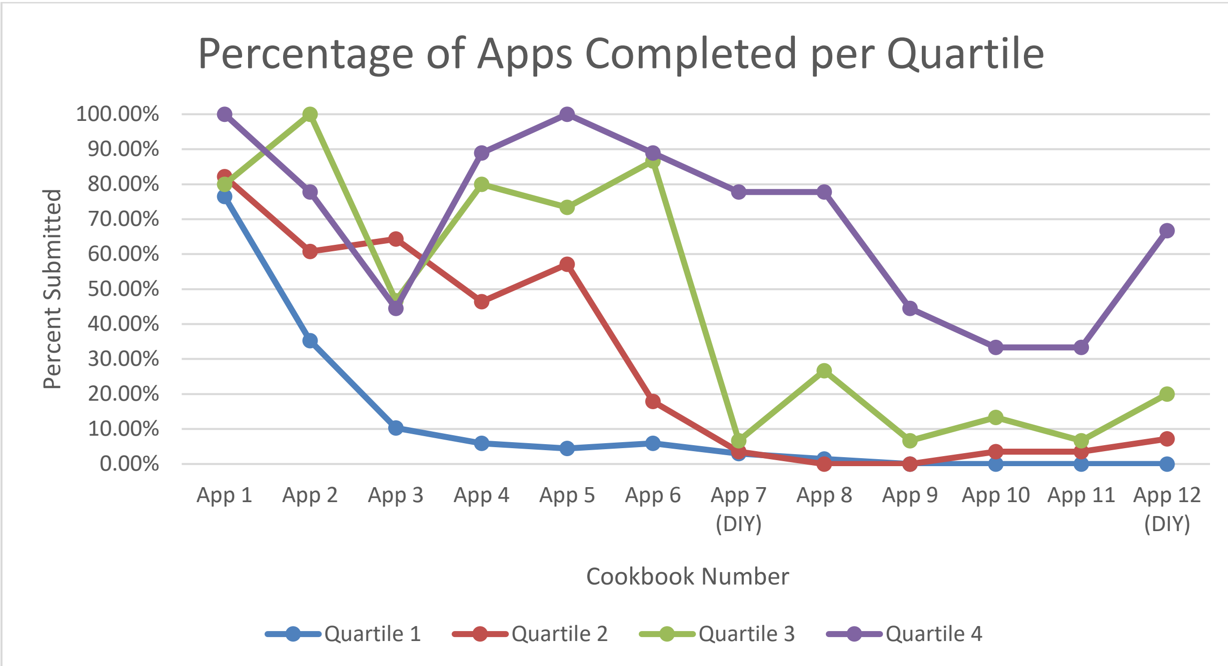

Figure 8. Percentage of students that completed each activity by quartile

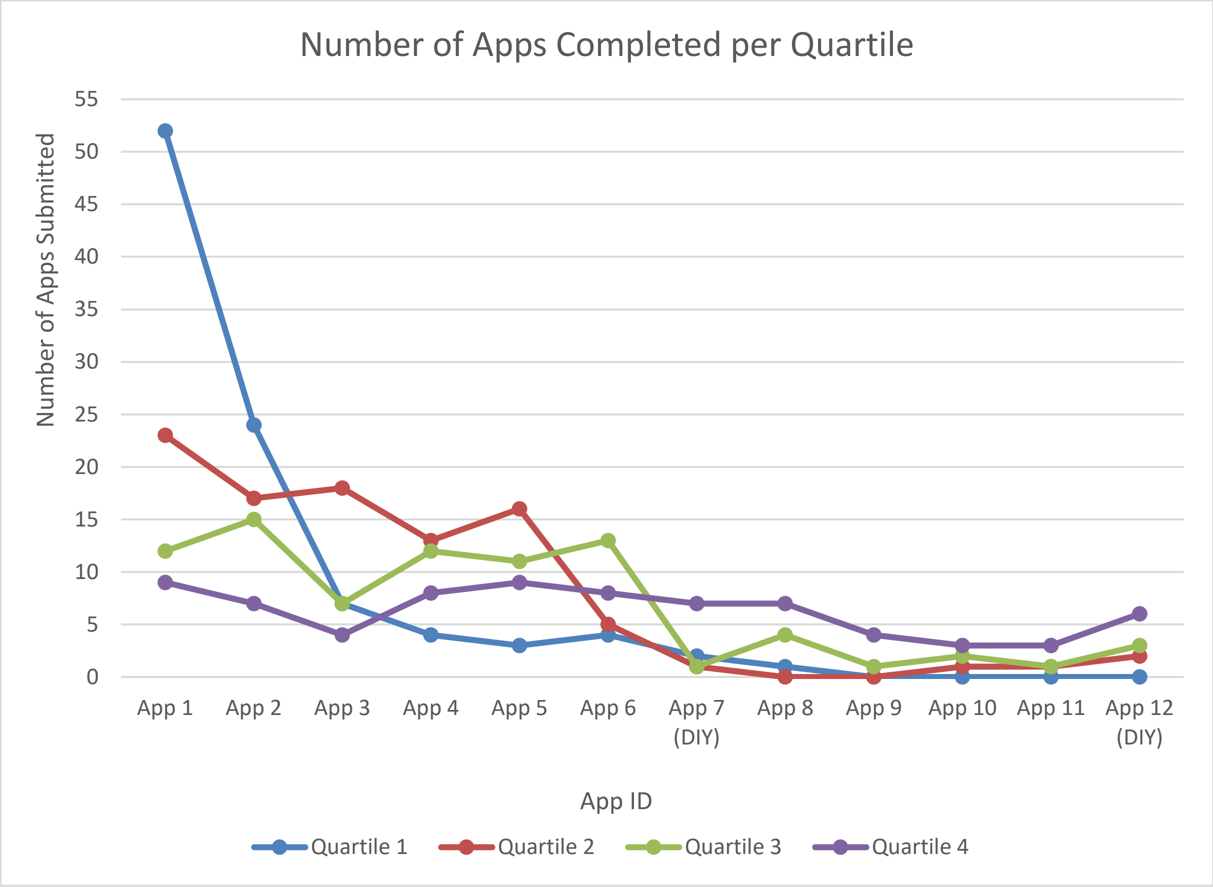

Figure 9. Number of apps submitted by each quartile

Students in the first quartile tended to only make the first two apps (Favorite Artist 1 and 2) though several students submitted apps 3-6. Most surprising is that student 648 went as far as to submit cookbooks 6 and 8, going the farthest in this quartile. It is unlikely that he skipped working on earlier projects and went straight to the more difficult cookbooks unless he had previous experience with App Inventor (which cannot be determined). Students 704 and 736 also have a similar situation in that they submitted a start to activity 7 (a DIY project with a health theme) that has some design work done (background color, page titles, a picture, and a textbox in one instance) but no coding. These three students did not submit any earlier activities, and these instances highlight the difficulty of definitively placing students into quartiles since it is likely that they would have worked on the earlier activities first, but we do not have a record of it.

Students in the 2nd quartile generally completed projects that ranged from Favorite Artist 1 to the Health DIY project (cookbooks 1-7), though one student got started on Space Invaders (cookbook 11) and another started LadyBug Chase (cookbook 10). Three participants in this quartile submitted DIY projects, with one working on the Health app (it has components arranged but no coding), another adding additional programming to her MLK app (activity 5) so that it would read the quotes out loud, and the last (student 715) completing a large DIY app promoting various musicians that spanned across multiple screens. Student 715 only completed the first two cookbooks (and started the 3rd cookbook without coding it), but her DIY project mostly employs techniques learned from the first two apps, so she was equipped to complete most elements of the app on her own. She may have had some assistance from an undergraduate mentor to figure out the navigation component. Her project represents our desire for students to apply what they learn into unique projects that are meaningful to them while also making efforts to learn new techniques either on their own or with the assistance of a peer or mentor.

Participants in the 3rd quartile completed projects across the spectrum of available activities, with a few starting (and one student completing) even the more difficult games. Four students from this quartile worked on DIY projects. The DIY projects included a health app (incomplete though it has listpickers and some code operating one of the listpickers), a remix of the MLK app with additional coding so that it will read the quotes out loud, a unique app that swaps out images when a button is pressed, and one project that appears to be the start of a game (due to the inclusion of sprites, balls, and a canvas) though it lacks code. Three out of four of the DIY projects have some coding incorporated into the app.

All of the participants in the 4th quartile made it to the more difficult apps (cookbooks 8-11) and eight of the nine students in the quartile attempted to create at least one DIY project. The

DIY projects (n = 12) include 2 remixed versions of the first cookbook, 7 health apps, one project that was the start of a game that a group was making at school 100, 1 modified version of app 11 (Space Invaders), and 2 completely unique projects. Student 507 was responsible for creating 3 DIY apps towards the end of the spring semester, including a version of Space Invaders with a design upgrade and additional programming (a background color was added and it was meant to change with a clock timer though there's an error in the code preventing the shift), a hypnotism app with flashing background colors and a rotating spiral (the error preventing the color change in the video game was corrected here), and a fan app for various celebrities (like the actress Zendaya) that was begun on the last day of the intervention.

CT Quiz

The CT quiz allowed researchers to see whether students understood the CT concepts that they learned from working on the cookbook activities and recognized their appearance in different coding scenarios. As discussed in chapter 3, participants in the top two quartiles had more exposure to the intervention's curriculum materials and this experience appears to explain why the 3rd and 4th quartiles had the strongest performance on the post-test of the CT exam. Table 43 contains participation data for the CT exam and the average post-test and change scores for the standard quartiles and for the quartiles with groupings that are based on the number of relevant apps completed.

Table 43

CT Quiz Data with Standard Quartiles and Quartiles Based on the Number of Relevant Apps Submitted

Quartile	Total Participants	Any Apps & 2+ CT Quizzes	CT Participation	Post-Test Avg.	Change Score Avg.
1	68	9	13.24%	23.15%	0.00%
2	28	8	28.57%	14.58%	1.04%
3	15	9	60.00%	43.52%	14.82%
4	9	6	66.67%	43.06%	20.83%
Quartile	Total Participants	Relevant Apps & 2+ CT Quizzes	CT Participation	Post-Test Avg.	Change Score Avg.
1	68	12	17.65%	19.44%	1.39%
2	28	8	28.57%	39.58%	14.59%
3	15	2	13.33%	37.50%	8.33%
4	9	6	66.67%	47.22%	20.83%

Ultimately, there was a difference between the quartiles based on their exposure to our curriculum and that had an effect on their demonstrated understanding of the CT concepts that the cookbook activities taught them. Participants in the fourth quartile did the most work and had the most frequent exposure to the more advanced CT concepts, so they outperformed their peers in the CT quiz and these students were more willing to engage with coding on their own in the DIY activities.

Quartile Comparison of Self-Efficacy, Affinity, Interest, and Persistence in ICT

Beyond the differences in the projects submitted and performance on the CT quiz, participant responses to questions on the profile surveys and interviews allow for comparisons to be made about their self-efficacy, their level of interest in app building/ICT, and their desire to continue learning about app building/ICT. In addition, more information about their previous experience and level of satisfaction with the intervention was gained. Tables 44-47 shows the average responses to the CT quiz post-test along with select questions from our other data collection methods in order to help illustrate the differences between the quartiles. Some of the data gives averages for the replies from the pre-test and post-test while other data only comes from one of the testing cycles; the source for each data set is indicated in each column heading. Tables 48-51 then ranks each quartile based on the participant responses to each question. The number one ranking quartile for each question/data source is highlighted in each column.

Table 44

Response Averages from CT Quiz, Surveys, and Interviews 1

Quartile	CT quiz performance (post-test)	ICT-21Q (post-test avg.)	SSS technology appreciation (post-test avg.)	SSS STEM careers appreciation (post-test avg.)	Previous app building experience (survey pre-test avg.)	Previous programming experience (survey pre-test avg.)
Q1	23.15%	3.51	5.45	5.43	2.12	2.00
Q2	14.58%	3.71	6.13	5.93	1.71	1.71
Q3	43.52%	3.81	5.38	5.82	1.13	1.88
Q4	43.06%	3.92	6.29	6.08	2.86	2.00

Table 45

Response Averages from CT Quiz, Surveys, and Interviews 2

Quartile	Level of confidence (survey pre-test avg.)	Felt good about what they were doing (survey post-test positive responses)	Were there any times when you felt like you were good at what you were doing? (interview positive response avg.)	Enjoyment of intervention (survey post-test avg.)	Enjoyment of intervention (fall/spring interview positive responses)	Enjoyment of intervention (spring interview only positive responses)
Q1	4.04	44.90%	70.00%	2.81	85.00%	100%
Q2	4.24	62.50%	75.00%	3.29	94.45%	100%
Q3	3.50	61.54%	72.86%	3.45	82.86%	85.71%
Q4	4.43	55.56%	100.00%	3.43	93.75%	100%

Table 46

Response Averages from CT Quiz, Surveys, and Interviews 3

Quartile	Collaboration (fall/spring interview positive response avg.)	Collaboration (spring interview positive response avg.)	Will create apps on one's own/outside of intervention (post-test, positive responses)	Likely to continue making apps (fall/spring interview positive response avg.)	Likely to continue making apps (spring interview only positive response avg.)
Q1	100%	100%	18.37%	75.00%	100%
Q2	83.34%	100%	29.17%	65.28%	75.00%
Q3	100%	100%	30.77%	48.57%	57.14%
Q4	100%	100%	33.33%	62.50%	100%

Table 47

Response Averages from CT Quiz, Surveys, and Interviews 4

Quartile	Would like to continue in the intervention (post-test survey, positive responses)	May continue in ICT programs (fall/spring interview positive response avg.)	May continue in ICT programs (spring interview only positive response avg.)
Q1	40.82%	80.00%	100%
Q2	58.33%	63.89%	50.00%
Q3	76.92%	62.86%	85.71%
Q4	66.67%	93.75%	100%

Table 48

Quartile Rankings Based on Average Responses 1

Quartile	CT quiz performance (post-test)	ICT-21Q (post-test avg.)	SSS technology appreciation (post-test avg.)	SSS STEM careers appreciation (post-test avg.)	Previous app building experience (survey pre-test avg.)	Previous programming experience (survey pre-test avg.)
Q1	3	4	3	4	2	2
Q2	4	3	2	2	3	4
Q3	1	2	4	3	4	3
Q4	2	1	1	1	1	1

Table 49

Quartile Rankings Based on Average Responses 2

Quar-tile	Level of confi-dence (survey pre-test avg.)	Felt good about what they were doing (survey post-test positive responses)	Were there any times when you felt like you were good at what you were doing? (interview posi-tive response avg.)	Enjoyment of intervention (survey post-test avg.)	Enjoyment of in-tervention (fall/spring interview positive response average)	Enjoyment of intervention (spring inter-view positive response average)
Q1	3	4	4	4	3	1
Q2	2	1	2	3	1	1
Q3	4	2	3	1	4	4
Q4	1	3	1	2	2	1

Table 50

Quartile Rankings Based on Average Responses 3

Quartile	Collaboration (fall/spring interview positive response average)	Collaboration (spring interview positive response average)	Will create apps on one's own/ outside of intervention (post-test, positive responses)	Likely to continue making apps (fall/spring inter-view positive response average)	Likely to continue making apps (spring interview positive response average)
Q1	1	1	4	1	1
Q2	4	1	3	2	3
Q3	1	1	2	4	4
Q4	1	1	1	3	1

Table 51

Quartile Rankings Based on Average Responses 4

Quartile	Would like to continue in the intervention (post-test survey, positive responses)	May continue in ICT programs (fall/spring interview positive response average)	May continue in ICT programs (spring interview positive response average)
Q1	4	2	1
Q2	3	3	4
Q3	1	4	3
Q4	2	1	1

The 4th quartile tended to rank the highest in most of the selected data sets: ICT 21Q, SSS technology and STEM careers appreciation, previous app building and programming experience, confidence going into the intervention, percentage of positive responses regarding times that they felt good at what they were doing (making/finishing apps was the most frequent response), likelihood in continuing to make apps (survey), and likelihood in participating in more ICT/app building programs in the future (interview).

The 3rd quartile had the highest performance on the CT quiz post-test, but the difference between their and quartile 4's average was less than one percent (a .46% difference). In addition, the 3rd quartile ranked highest in their enjoyment of the intervention on the survey and desire to continue in the intervention on the survey (though they were ranked last in the interviews on these questions when looking at the interview feedback). The 3rd quartile was a high achieving group in terms of their app submissions and scores on the CT post-test, but their enthusiasm for the program seems appears to be mostly mixed with more positive attitudes towards app building/ICT shown on the survey but not as much comparatively on the interviews.

The 2nd quartile ranked highly in their enjoyment of the intervention (interview average across the year though they ranked 3rd on the survey post-test) and in their indication that they felt good about things they were doing in the intervention (ranked 1st in the survey and 2nd in the interview). These students did appear to get something out of the program though they may not necessarily continue to build apps or participate in other ICT interventions.

The 1st quartile did not rank the highest for any of the data sets, but there were instances where some of the quartiles tied for first—collaboration and several interview responses when focusing only on the spring semester (enjoyment of the intervention, likelihood of continuing to make apps, and likelihood of participating in another ICT program). Data isolating the spring semester interviews was included to show the student's mindset at the end of the year, and quartiles 1 and 4 overlapped the most in these instances.

Self-Efficacy in ICT

Students in the higher quartile appear to have a stronger belief in their ability to work in ICT/app building based on responses to questions from the profile surveys and interviews. The ICT-21Q is one of the most direct measures of self-efficacy that this study used as it asks students to rate their ability with various 21st century skills like their ability to write code and to understand and use technology systems and technology application. While the average scores on the ICT-21Q decreased for the first two quartiles from pre-test to post-test, the average scores increased for the upper quartiles (these results are discussed in detail in chapter 3). Table 52 focuses on the results for 21st century skills that were prevalent in the intervention: the ability to work with technology systems, technology applications, and code. The changes in the average scores show consistent increases for the 4th quartile while the other quartiles had mixed results. The 3rd quartile's responses were interesting in that they may have interpreted the skill of understanding and using technology applications as specifically being about app building or using App Inventor, and their scores declined but, nevertheless, their scores increased for writing code. This quartile had the highest performance on the CT exam, so their belief in their ability in that regard is not necessarily at question, but the students may have less optimism about using App Inventor specifically.

Table 52

Profile Survey Averages for Select Skills from the ICT-21Q

Quartile	Understand and use technical systems			Understand and use technology applications			Write code		
	Pre	Post	Change	Pre	Post	Change	Pre	Post	Change
1	4.00	3.50	-0.50	4.21	3.36	-0.85	3.86	3.43	-0.43
2	4.08	3.92	-0.16	3.46	3.46	0.00	3.62	3.38	-0.24
3	3.10	3.44	0.34	3.70	3.00	-0.70	3.00	3.11	0.11
4	3.13	4.00	0.87	3.63	3.75	0.12	3.38	3.57	0.19

During the pre-test of the profile survey, students in the 4th quartile had the most confidence going into the intervention (4.43 average on a scale 1-5) though students in the 3rd quartile had the least confidence (3.50). This seems to indicate that confidence going in does not necessarily determine how much the students will achieve, but the students in the top quartile nevertheless had the most confidence on average. Having previous experience with app building or programming produced similar results, with quartile 4 having the highest percentage of responders stating that they had previous experience in these areas (71.43% and 100% respectively), showing that some previous experience in app building and coding may have helped spur them to progress the farthest in the intervention as well as come in with the most confidence. However, students in quartile 3 had the lowest percentage of experience in app building (12.50%) and a close percentage of experience in programming to the 2nd quartile (Q3, 37.50% vs. Q2, 35.29%), so high achievement is not entirely dependent on experience or confidence though students in the 4th quartile had the highest averages in these areas.

Lastly, the surveys short answer questions and the interviews revealed that most responders to the interviews and surveys indicated that learning to build apps or just being part of the intervention made them feel good or was the most important thing that they learned. Some students in the first 3 quartiles mentioned that they became more confident during the semester and the majority of interview responders were interested in sharing their apps online. There was not a noticeably large difference between the quartiles in terms of their interest in building apps or likelihood in seeing value in learning the skill.

Affinity, Interest, and Persistence in ICT

The surveys and interviews give the clearest indication of whether interest in app building/ICT may have changed over the course of the intervention for students in each quartile. The STEM Semantic Survey asked students to choose between a series of descriptors on a Likert scale (from 1 to 7) in order to gauge their interest in Technology and STEM careers, and the 4th quartile's average scores rose the highest (from 5.97 to 6.29) from pre-test to post-test, giving them the most positive view on the topic in the post-test. Averages diminished for students in quartiles 1 and 3 (declining by .35 and .71 points respectively) while quartile 2 increased slightly (by .06 points). The changes in scores indicate that the students relative interest in technology did not necessarily boost or inhibit performance since students in the top two quartiles had the highest and lowest average scores in their opinion about technology. Nevertheless, students in the 4th quartile had the highest average. The participants opinion about STEM careers differed from their opinion about technology in that each quartile increased their average scores from pre-test to post-test except the 4th quartile (diminishing by .01 points). The 4th quartile still had the highest average, the 1st quartile had the lowest average, and the 2nd and 3rd quartiles averages differed slightly in the post-test (Q1: 5.43; Q2: 5.93; Q3: 5.82; Q4: 6.08). With the 2nd quartile having a higher average from the 3rd quartile, there is some indication that one's view on a career in STEM did not necessarily help or hinder performance in the intervention. Nevertheless, students in the 4th quartile had the most positive attitude towards STEM careers.

The post-test interview directly asked students to rank how much they liked what they were doing on a scale from 1-4, and the 3rd and 4th quartiles had the highest average scores (Q1: 2.89; Q2: 3.29; Q3: 3.45; Q4: 3.43), indicating that the students that did the most work had the highest level of enjoyment. Furthermore, students in the 3rd and 4th quartile were more likely to indicate that they wanted to continue in similar interventions or build apps on their own. Ultimately, the students in the 4th quartile arguably had the most positive opinion about app building/ICT while the 3rd quartile may have largely had positive opinions about app building/ICT but they did not always surpass the 2nd quartile in their average responses. The 1st quartile tended to rank last in most measurements.

Conclusions

Ultimately, the intervention can claim success at teaching CT skills if students worked through enough of our cookbook activities since the top quartiles that did the most work had the best performance on the CT quiz and made more efforts to branch out and explore with app development by working on DIY projects while some students in the lower 3 quartiles tended to express more doubt in their ability to work on a DIY app (though some students did want to make DIY apps). Scaffolding and support from the researchers, mentors, and other participants did help students proceed through the material and proved to be one important aspect to the intervention though some students had some friction with school faculty, which adversely effected the community building that was occurring at some of the schools.

The data on self-efficacy shows that students in the 4th quartile had the highest average scores regarding their belief in their ability to understand and use various 21st century skills, particularly skills relevant to coding and the development of apps. It can be argued that high self-efficacy was linked to accomplishment in the intervention, but the difference for average scores of self-efficacy between quartiles was not statistically significant. Therefore, a firm affirmation of the importance of self-efficacy cannot be made although the top quartile had the highest average.

Lastly, students in the 4^{th} quartile had the highest level of interest in app building/ICT and are the most likely to continue learning and engaging in this content matter. Differences in opinion between the 2^{nd} and 3^{rd} quartiles differed and a conclusive distinction between the two may be difficult to make, especially since some data sets that ask similar questions had conflicting responses (for instance, students in the 3^{rd} quartile outranked the other quartiles in their enjoyment of the intervention during the survey but not during the interview as one student gave more negative or neutral responses in those contexts and the low sample size weighed those responses quite heavily though the student was an outlier). The lack of a clear distinction between the 2^{nd} and 3^{rd} quartiles may be due to some students in the 3^{rd} quartile being compelled to have attended and not being entirely happy about the circumstance despite completing a lot of work. Larger sample sizes for each quartile might have delineated the quartiles much more.

Implications

The structure of this study and its findings align with those of other researchers. Scholars have pointed out that lack of early exposure and preparation in ICT/STEM fields puts URM students at a disadvantage (Strayhorn, 2015). Many scholars champion the importance of early preparation, including informal, workshop environments or after-school programs that can fill in gaps that are not met in the school day, like teaching computer science/ICT curriculum (Denson, Hailey, Stallworth, & Householder, 2015; Foltz, Gannon, & Kirschmann, 2014; Mouza, Marzocchi, Pan, & Pollock, 2016; Palmer, Maramba, & Daney, 2011; Sahin, 2013). In these learning experiences, the importance of turning students into producers of media and information rather than just consumers of media (Gretter & Yadav, 2016; Kafai & Burke, 2014), providing mentors that care about the student's success (Campbell et al., 2014; Duran, Höft, Lawson, Medjahed, and Orady, 2014; Meador, 2018), and having students actively involved in authentic tasks and creating products for a real audience (Blumenfeld, Soloway, Marx, Krajcik, Guzdial, and Palincsar, 1991; Roberts, Jackson, Mohr-Schroeder, Bush, Maiorca, Cavalcanti, Schroeder, Delaney, Putnam, & Cremeans, 2018) have been discussed and help compose the core of this research project.

On the topic of turning students into producers of media, most students that were interviewed or responded in the profile survey enjoyed making apps. The majority of students that were interviewed felt that making apps was the coolest thing that they learned in the intervention and they were willing to continue in similar activities (though an equal number of students in quartile 2 were uncertain if they would join additional activities) or make apps on their own (though a nearly equal number of students in quartile 3 expressed disinterest or uncertainty about making apps on their own in the spring). The majority of responses to interviews in the fall spring indicated that the students felt good about making apps. One spring student mentioned wanting to share apps online whereas many more students in the fall discussed being interested in sharing their apps except one student in quartile 3 expressing doubt that others would like the apps. In the survey, participants in quartiles 3 and 4 felt more positive about what they were doing in the intervention and were more inclined to continue with ICT activities or app building on their own. More data specifically about sharing apps may need to be collected to make a stronger connection, but there was a high percentage of interest in app building among the sample of students that responded to the interviews and surveys.

During the pre-test of the profile survey pre-test, we gathered data about the student's background with app building and general programming experience. This data does not reveal the nature of their previous experience, which could be a one-time opportunity, recreational activity (like using LEGO K'NEX), participation in an after-school activity, or something else entirely. Table 53 shows the results from the survey.

Table 53

Previous App Building and Programming Experience by Quartile

Quartile	*n*	I feel confident in my ability to learn how to build phone apps.	Have you built an app before?	Have you ever tried to write computer programs or coded before?
1	28	4.04	50.00%	57.14%
2	17	4.24	35.29%	35.29%
3	8	3.50	12.50%	37.50%
4	7	4.43	71.43%	100.00%

The profile survey's pre-test revealed that students in the 4th quartile had the most experience in app building and programming, which most likely came from using other platforms (like Scratch) or participating in other after-school activities (like robotics). Previous experience may be a determining factor in their persistence, but more data would need to be gathered to help differentiate the results for quartiles 2 and 3 since students in the 3rd quartile surpassed the 2nd in activity submission though considerably fewer had previous app building experience and only slightly more had programming/coding experience. It is also interesting that half of the responders in quartile 1 had some form of previous experience, but their attrition may have been predicated on any number of factors (like participation in other activities) and not necessarily difficulty or frustration with the curriculum.

On the issue of providing mentors that care about success, scholars have brought up the importance of having role models that match the ethnicity or gender of students (Hill, Corbett, and St. Rose, 2010; McGee, 2015). In this case, the majority of the on-site research team and all of the undergraduate computer science mentors identified as black/African American, which in turn reflected the majority of the population attending the after-school intervention, allowing mentors and researchers to serve as role models and demonstrating that continued study and participation in ICT/STEM is for everyone. Furthermore, findings from this study point out the importance of scaffolding from the mentors and researchers to help students get used to the App Inventor platform and become more confident in their work. In the fall semester, 50-55% of students in quartiles 1 and 4 mentioned that they collaborated with a mentor (with 11% and 25% in quartiles 2 and 3 stating so respectively). However, between 66% and 100% of students in quartiles 2-4 mentioned that they worked with a mentor during the spring, so this aspect of the instructional design should continue in future iterations as students became more cognizant of its importance to themselves.

On the topic of task authenticity and creation of products for a real audience, the use of App Inventor makes the authenticity of what participants are doing immediately obvious since the platform creates apps that can be shared on Android phones. Furthermore, students can create an app relatively quickly given platform's use of block-based coding, a form of visual programming language. Researchers have discussed the utility of visual programming languages such as Scratch and App Inventor for introducing students to computational thinking concepts (Maloney, Peppler, Kafai, Resnick, & Rusk, 2008; Sáez-López, Román-González, & Vázquez-Cano, 2016) and even as a springboard for learning more programming languages (Armoni, Meerbaum-Salant, & Ben-Ari, 2015). Papadakis, Kalogiannakis, Orfanakis, and Zaranis (2014) discussed the advantages of App Inventor over other block-based coding environments due to its ability to create apps usable on phones and its more expansive capabilities for programming over other VPLs like Scratch. The effort of this instructional design, similar to that of other researchers, sought to use a VPL (App Inventor) in an after-school workshop environment in order to give students hands-on experience creating apps that can be used on Android phones. The immediate utility of the finished product and the support system was intended to prepare students to create artifacts and keep the instructional design in line with other scholars.

Unfortunately, this study does not have a focus on creating apps for a real client. The National Academies of Sciences, Engineering, and Medicine point to the importance of engaging students in problem-solving, team-based activities, and peer support (along with tutoring and a flipped classroom design) as ways to redesign introductory STEM courses in order to support student success rather than weed them out while also bringing social elements (like a sense of belonging) into classes. Some universities, like the University of Maryland, Baltimore County, specifically followed guidelines for an active learning approach with some professors citing success with less class failures and more students majoring in various STEM fields (Hrabowski & Henderson, 2017, p. 35). Though shortcomings in the pilot study stemmed from students not persisting through the 2nd semester (when the team project-based learning activities were underway), the intervention team should return to the format in future iterations with greater resolve due to the observed importance of peer collaboration among the students (including their own recommendations for it) and the potential for team problem-solving activities to get students to build more experimental, DIY apps rather than stick to the cookbooks.

Though project-based learning (PBL) activities worked for various researchers, other researchers, like Maloney et al. (2008), tend to gravitate more towards allowing students to work on individual projects while avoiding explicit instruction. This intervention can use both formats in future iterations, but we should consider adapting the idea of Maloney et al. (2008) that involves incorporating periodic marathons where students work on a single DIY project for 3-4 hours and then share the results with the other club members (p. 368). The marathon could, perhaps, be framed as a competition, which is an idea we also wanted to implement during the pilot study but could not logistically bring students from disparate sites to one location for the event. Instead, we encouraged students to work in small teams and prepare an app and website for a regional science fair competition held during the spring semester. This was of interest to some of the students, but attrition in the spring for various reasons (like tutoring and other activities) made it impractical to get a completed project submitted. In future iterations of the intervention, it might be possible to have our own online competition hosted in-house by having students work on a single project for 1 week, submit it to the intervention's LMS along with a description and purpose statement for the project (written or filmed), and then have either a panel of experts or a poll determine the winner.

Expanding beyond the issue of including group work is the underlying focus of community building, not just among intervention participants and staff but also the broader community. Lachney (2017) and Scott, Sheridan, and Clark (2015) stress the importance of building assets for the broader community, so some collaboration with members of the community acting as clients and ITC/CS students acting as developers could be arranged. Community leaders, charities, foundations, educators, or some other agents could be found that have basic needs for an app to help their organization or specific project, and teams of participants of the after-school app building intervention could compete to create a product that meets the client's needs or the teams could collaborate and compartmentalize the development of a much bigger app, with teams dividing up responsibilities. For instance, one team could be responsible for learning to program every aspect of the app that deals with data storage while another team could work on designing and laying out every screen of the app. The issues would be to make sure the demands of the client are practical and within the ability of our participants to complete.

Adding team-based, PBL activities either in the form of a completion or to help meet the needs of a real client could serve as a means for building interest in the students and showing the practical utility in what they are learning. Though adding more emphasis on DIY projects for helping the community would be a good addition to the program, but students still need to have time at the beginning of the school year to familiarize themselves with the App Inventor platform and learn about computational thinking. Ryoo et al. (2013) may have the best framework to consider, suggesting that a PBL take place as a later module following implementation of knowledge building activities.

In addition to community building, many scholars note the importance of making classroom activities culturally relevant to the learners. Participants in this intervention stated their appreciation for apps like Favorite Artist and MLK because of their cultural appeal, and even the Health DIY app appealed to some students because it put more control in their hands and because it could help others. In addition, many students stated in the interviews that they would like to share their apps online for others to use, so there is already a desire to make meaningful apps that can interest a broader audience. The intervention LMS did contain some themed DIY projects that could have broader appeal (such as an app that can send a student’s location to his or her parents, an app about endangered species, and a virtual fieldtrip of one’s school), but we will need to encourage the completion of such projects more strongly.

As noted in chapter 4, one student recommended that we give the students raw code and let them figure out how to apply it in their own work. Creating a database with algorithms and tips might be a useful resource to add to the intervention's LMS, especially if we allow the students to contribute their own code strings and explanations. Furthermore, giving students a worksheet with a glossary of terms (like the names of the most commonly used CT concepts) might be another helpful resource to help students articulate their practices. All of this might be useful for spurring them to create more DIY activities and think more like CS experts.

Lastly, the intervention may need to require students to submit their work before leaving each period in order to serve as a backup should their files be lost and just to ensure that we are receiving copies of everything students complete. Another idea would be to incorporate a ticketing system where students alert the LMS with the app they are starting and cannot receive instructions for a new app without submitting a project or a notice (including a brief rationale) that they are switching projects. Alternatively, we could also have students log their activities before leaving the classroom/computer lab (what project are they working on, did they work with a peer, when they expect to finish, and whether they need help with anything).

Suggestions for Further Research

Future studies for a similar after-school program could expand upon the findings for this study by testing the intervention in different contexts, collecting more data on self-efficacy, gathering data on how competition effects motivation, changing data collection methods, exploring the effect of parental and community involvement, and performing a longitudinal study to see what students do next and if they see a connection between the CT concepts and practices that they learn in the intervention and how they can be applied in other contexts.

Testing in different contexts

Data was very difficult to collect given the context of being in an after-school program. Students could join at any point, leave at will, or be pulled from participating due to tutoring, standardized testing, practice for sports, band, and dance, moving from the school district, family responsibilities, and any host of other reasons since the intervention was not mandatory as normal classes and activities would be during the school day. Some students even recommended that we hold the intervention during school hours as a normal class, which would have made students more responsive to participating in all of our data collection efforts. The intervention could also be tested as an intensive summer program where students meet for longer periods of time though over a shorter timespan (4-7 weeks rather than a full year). Lastly, the intervention might also have different results if tested with students in different locations (suburban, country, etc.) besides just a major metropolitan region, which would be interesting for a comparison.

Collect more data on self-efficacy

Each of the quartiles had average scores on the ICT-21Q portion of the profile survey that were close though the 4th quartile had the highest average on the post-test. The survey and interviews mostly lacked follow up questions explicitly asking students about their confidence related to app building or coding more specifically. The profile survey pre-test did include one additional question specific to self-efficacy, “I feel confident in my ability to learn how to build phone apps,” but a similarly worded question was not included in the post-test for a comparison after having participated in the intervention. This is additional data that can help supplement the ICT-21Q, and similar interview questions should also be included about the topic.

Wang (2013) found self-efficacy to be an influence among URM and a factor that could determine one’s pursuit of a STEM degree. Olszewski-Kubilius, Steenbergen-Hu, Thomson, and Rosen (2016) also found that STEM self-efficacy better predicted student’s intentions to persist in STEM after instruction (p. 30). However, Andersen and Ward (2014) did not find self-efficacy to be a significant predictor for URM students because the questions revolved around perception of ability to succeed in 9th grade coursework rather than in math and science in general or in higher level coursework (p. 231). Data gathered in this intervention would be more about coding and app building in general and not tied to school performance, and it would be interesting to see how students feel about their ability to code and build apps as a hobby and as a possible career option. Determining if self-efficacy can predict future pursuit is difficult, but it may be a predictor for how much work one does in the intervention.

Effect of file sharing and competiton

As discussed, team-based activity could be a very beneficial for building motivation and a sense of community among the students. Encouraging students to publish their original apps on App Inventor to enter the App of the week/month contest could also be one source of motivation for them as well. Whether a programming marathon (i.e. hackathon) with a competitive element is advisable or not, file sharing is something of interest to most of the students interviewed in this intervention and could be an interesting point of data collection that could boost motivation and even self-efficacy if participants are subjected to direct feedback (provided they receive positive reviews or constructive criticism). If students build an app for a client, then they could receive formal, written evaluation about the app's success in meeting the client's objectives. Measuring the effect of feedback can be determined through post-test survey and interview questions.

If students can opt to have their apps published on the intervention's LMS, we could allow their peers to leave reviews in a comments section beneath or on the LMS forum, which could provide valuable feedback to the creators and even lead to new versions of apps with improved features being created. Teams or individuals that publish new versions can be encouraged to write version histories with a list of the updated features and bug fixes to help market why users will want to download the new version rather than just continue using the old. Other practices used by expert app and software developers can also be researched and employed to further teach best practices.

In addition to giving students more opportunities to thoughtfully give and receive feedback to their peers, there is an opportunity to get students to engage in more critical analysis of their own work. Students can be asked to provide a rationale for their content and design choices in their DIY projects. Gretter and Yadav (2016) particularly champion having an element of critical thinking/analysis due to the importance of being able to understand how content is created, how to evaluate it, and how to use it meaningfully (p. 512). Critical thinking is also one of the points of focus of the Partnership for 21st Century Learning (Battelle for Kids, n. d.), and a point of interest that could be explored more closely by looking at samples of student's own critiques, their responses to feedback that they receive, and their reflective writing regarding their own apps.

Changing data collection methods

Changing the way data is collected is an important way to help us gather data from more participants. Several students expressed dislike of the long profile surveys during the interviews and as feedback during the surveys, nothing them as their least favorite part of the intervention due to their length and the frequency of all the online data collection efforts (also including the CT quiz). Since students did not largely contribute data to us through these means, researchers could obtain feedback from more participants on a daily basis with just a short, satisfaction survey at the end of each period where they tell us what they did during the period, express their level of confidence, and state their level of interest. Additionally, the ICT-21Q and SSS could still be included as part of a pre-test/post-test design, but we should edit out the unnecessary questions (like whether they believe IT professionals use each 21st century skill and their level of interest in Science, Engineering, and Math) and keep the survey more focused on getting feedback in very specific areas (interest, self-efficacy, and satisfaction with the program) while leaving off extraneous material.

A longitudinal study on what students do next

Students in the 4th quartile mostly had prior experience with programming, but this after-school intervention was the first opportunity that some students had with coding. While the extent of the student's prior experience cannot be determined for the current study, it would be an interesting part of data collection to get more detailed information about the students' backgrounds with programming and to track what students do after completing the intervention (do they take similar CS/ICT programs or classes in high school)? It would also be interesting to know if students can apply what they learned about CT in the intervention to other contexts, like in a robotics club or a CS class where they learn standard programming languages like Java, Python, or C++. Feedback from surveys or interviews could reveal if students felt a connection between the intervention and other ICT/CS programs they have attended, but monitoring will be challenging if students frequently switch schools, stop attending after-school programs, or refuse to give assent to prolonged data collection efforts.

REFERENCES

American Immigration Council (2017, June 4). Fact sheet: Foreign-born STEM workers in the United States. Retrieved from https://www.americanimmigrationcouncil.org/research/foreign-born-stem-workers-united-states

Andersen, L., & Ward, T. J. (2014). Expectancy-value models for the STEM persistence plans of ninth-grade, high-ability students: A comparison between Black, Hispanic, and White students. *Science Education*, *98*(2), 216-242. doi:10.1002/sce.21092

Armoni, M., Meerbaum-Salant, O., & Ben-Ari, M. (2015). From Scratch to "real" programming. *ACM Transactions on Computing Education (TOCE)*, *14*(4), 1-15.

Aronson, B., & Laughter, J. (2016). The theory and practice of culturally relevant education. *Review of Educational Research*, *86*(1), 163-206. doi:10.3102/0034654315582066

Aspray, W., & Bernat, A. (2000, March). Recruitment and retention of underrepresented minority graduate students in computer science. In *Report on a Workshop by the Coalition to Diversity Computing*.

Ayer, T., Cohen, C. & Calandra, B. (accepted, 2018). Computational Thinking Instrument Validation. Paper presented at the American Education Research Association (AERA) 2018 conference, New York, NY.

Bandura, A. (1995). *Self-efficacy in changing societies*. Cambridge, UK: Cambridge University Press.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community? *ACM Inroads*, *2*(1), 48-54.

Battelle for Kids (n. d.). P21 network. Retrieved from http://www.battelleforkids.org/networks/p21

Bielaczyc, K. (2006). Designing social infrastructure: Critical issues in creating learning environments with technology. *The Journal of the Learning Sciences*, *15*(3), 301-329.

Blumenfeld, P. C., Soloway, E., Marx, R. W., Krajcik, J. S., Guzdial, M., & Palincsar, A. (1991). Motivating project-based learning: Sustaining the doing, supporting the learning. *Educational Psychologist*, *26*(3-4), 369-398.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *AERA 2012*. *Proceedings of the 2012 annual meeting of the American Educational Research Association (pp. 1-25).* Vancouver, BC, Canada: American Educational Research Association. Retrieved from http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf

Brown, M., Dehoney, J., & Millichap, N. (2015, April). The next generation digital learning environment: A report on research. *EDUCAUSE Learning Initiative*. Louisville, CO: EDUCAUSE.

Brown, P. L., Concannon, J. P., Marx, D., Donaldson, C. W., & Black, A. (2016). An examination of middle school students' STEM self-efficacy with relation to interest and perceptions of STEM. *Journal of STEM Education: Innovations & Research*, *17*(3), 27-38.

Bureau of Labor Statistics (n.d.-a). Table 2: Employment status of the civilian noninstitutional population 16 years and over by sex, 1975 to date. In *2015 annual averages – household data –tables from employment and earnings*. Retrieved from https://www.bls.gov/cps/cpsa2015.pdf

Bureau of Labor Statistics (n.d.-b). Table 3: Employment status of the civilian noninstitutional population by age, sex, and race. In *2015 annual averages – household data –tables from employment and earnings*. Retrieved from https://www.bls.gov/cps/cpsa2015.pdf

Bureau of Labor Statistics (n.d.-c). Table 4: Employment status of the Hispanic or Latino population by age and sex. In *2015 annual averages – household data –tables from employment and earnings*. Retrieved from https://www.bls.gov/cps/cpsa2015.pdf

Bureau of Labor Statistics (n.d.-d). Table 12: Employed persons by sex, occupation, class of worker, full- or part-time status, and race. In *2015 annual averages – household data – tables from employment and earnings*. Retrieved from https://www.bls.gov/cps/cpsa2015.pdf

Burke, K., & Dunn, R. (2003). Learning style-based teaching to raise minority student test scores: There’s no debate! *The Social Studies*, *94*(4), 167-170.

Burnett, M. M. (1999). Visual programming. *Wiley Encyclopedia of Electrical and Electronics Engineering*. J. G. Webster (Ed.). doi:10.1002/047134608X.W1707

Calandra, B., Renken, M., Cohen, J., Ayer, T., & Hicks, T. A. (in progress). An examination of a group middle school students’ engagement and computational thinking outcomes during a series of afterschool computing activities.

Campbell, A. G., Skvirsky, R., Wortis, H., Thomas, S., Kawachi, I., & Hohmann, C. (2014). NEST 2014: Views from the trainees—Talking about what matters in efforts to diversify the STEM workforce. *Cell Biology Education*, *13*(4): 587- 592.

Charmaz, K. (1996). The search for meaning – Grounded theory. In J. A. Smith, R. Harre, & L. Van Langenhove (Eds.), *Rethinking methods in psychology* (pp. 27-49). London: Sage Publications.

Chen, X., & Soldner, M. (2013, November). STEM attrition: College students' paths into and out of STEM fields. Retrieved from https://nces.ed.gov/pubs2014/2014001rev.pdf

Choi, Y. (2013). Teaching social studies for newcomer English language learners: Toward culturally relevant pedagogy. *Multicultural Perspectives*, *15*(1), 12-18. doi:10.1080/15210960.2013.754640

Cohen, J. D., Renken, M., & Calandra, B. (2017). Urban middle school students, twenty-first century skills, and STEM-ICT careers: Selected findings from a front-end analysis. *TechTrends*, *61*(4), 380-385.

Creswell, J. W. (2005). *Educational research: Planning, conducting, and evaluating quantitative* and qualitative research (2nd ed.). Upper Saddle River, NJ: Pearson Prentice Hall.

Creswell, J. W., & Creswell, J. D. (2005). Mixed methods research: Developments, debates, and dilemmas. *Research in organizations: Foundations and methods of inquiry*, 315-326.

Delacre, M., Leys, C., Mora, Y. L., & Lakens, D. (2019). Taking parametric assumptions seriously: Arguments for the use of Welch's F-test instead of the classical F-test in one-way ANOVA. *International Review of Social Psychology*, *32*(1), 1-12.

Denson, C. D., Hailey, C., Stallworth, C. A., & Householder, D. L. (2015). Benefits of informal learning environments: A focused examination of STEM-based program environments. *Journal of STEM Education: Innovations and Research*, *16*(1), 11-15.

Design-Based Research Collective. (2003). Design-based research: An emerging paradigm for educational inquiry. *Educational Researcher*, *32*(1), 5-8.

Domínguez, A., Saenz-De-Navarrete, J., De-Marcos, L., Fernández-Sanz, L., Pagés, C., & Martínez-Herráiz, J. J. (2013). Gamifying learning experiences: Practical implications and outcomes. *Computers & Education*, *63*, 380-392.

Duran, M., Höft, M., Lawson, D., Medjahed, B., & Orady, E. (2014). Urban high school students' IT/STEM learning: Findings from a collaborative inquiry- and design-based afterschool program. *Journal of Science Education & Technology*, *23*(1), 116-137. doi:10.1007/s10956-013-9457-5

Ericson, B., & McKlin, T. (2012). Effective and sustainable computing summer camps. In *SIGCSE 2012. Proceedings of the 43rd ACM Technical Symposium on Computer Science Education* (pp. 290-294). New York, NY: Association for Computing Machinery.

Fayer, S., Lacey, A., & Watson, A. (2017, January). STEM Occupations: Past, present, and future. U. S. Bureau of Labor Statistics. Spotlight on statistics. Washington, DC: United States Department of Labor, U. S. Bureau of Labor Statistics. Retrieved from https://www.bls.gov/spotlight/2017/science-technology-engineering-and-mathematics-stem-occupations-past-present-and-future/pdf/science-technology-engineering-and-mathematics-stem-occupations-past-present-and-future.pdf

Foltz, L. G., Gannon, S., & Kirschmann, S. L. (2014). Factors that contribute to the persistence of minority students in STEM Fields. *Planning for Higher Education*, *42*(4), 1-13.

Games, P. A., & Howell, J. F. (1976). Pairwise multiple comparison procedures with unequal n's and/or variances: A Monte Carlo study. *Journal of Educational Statistics*, *1*(2), 113-125.

Garrison, D. R. (1997). Self-directed learning: Toward a comprehensive model. *Adult Education Quarterly, 48*(1), 18-33.

Gastwirth, J. L., Gel, Y. R., & Miao, W. (2009). The impact of Levene's test of equality of variances on statistical theory and practice. *Statistical Science*, *24*(3), 343-360.

Gay, G. (2013). Teaching to and through cultural diversity. *Curriculum Inquiry*, *43*(1), 48-70. doi:10.1111/curi.12002

Glaser, B. G., & Strauss, A. L. (1967). The discovery of grounded theory: Strategies for qualitative research. *New Brunswick: AldineTransaction*.

Glen, S. (n.d.). F statistic / F value: Simple definition and interpretation. *Statistics How To*. Retrieved from https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/f-statistic-value-test/

Greene, J. C., Caracelli, V. J., & Graham, W. F. (1989). Toward a conceptual framework for mixed-method evaluation designs. *Educational Evaluation and Policy Analysis, 11*(3), 255-274.

Gretter, S., & Yadav, A. (2016). Computational thinking and media & information literacy: An integrated approach to teaching twenty-first century skills. *TechTrends*, *60*(5), 510-516.

Grow, G. O. (1991). Teaching learners to be self-directed. *Adult Education Quarterly*, *41*(3), 125-149.

Guglielmino, P. J., Guglielmino, L. M., & Long, H. B. (1987). Self-directed learning readiness and performance in the workplace: Implications for business, industry, and higher education. *Higher Education*, *16*, 303-317.

Gurin, P. (1999). Expert report of Patricia Gurin. *Michigan Journal of Race and Law*, *5*(1), 363-425.

Hanson, W. E., Creswell, J. W., Clark, V. L. P., Petska, K. S., & Creswell, J. D. (2005). Mixed methods research designs in counseling psychology. *Journal of Counseling Psychology*, *52*(2), 224-235.

Heck, I. (31 Oct 2013). Lacking foundation, minorities struggle in STEM fields. *The Brown Daily Herald*. Retrieved from http://www.browndailyherald.com/2013/10/31/lacking-foundation-minorities-struggle-stem-fields/

Hill, C., Corbett, C., & St. Rose, A. (2010). *Why so few? Women in science, technology, engineering, and mathematics*. Washington, DC: American Association of University Women.

Homack, S. R. (2001, February 1). *Understanding what ANOVA post hoc tests are, really*. Paper presented at the Annual Meeting of the Southwest Educational Research Association (pp. 1-15). New Orleans, LA: Southwest Educational Research Association. Retrieved from https://files.eric.ed.gov/fulltext/ED449222.pdf

Hrabowski III, F. A., & Henderson, P. H. (2017). Toward a more diverse research community: Models of success. *Issues in Science and Technology*, *33*(3), 33-40.

IBM. (n.d.). *GLM Univariate Analysis*. IBM Knowledge Center. Retrieved from https://www.ibm.com/support/knowledge-center/en/SSLVMB_23.0.0/spss/base/idh_glmu.html

Intemann, K. (2009). Why diversity matters: Understanding and applying the diversity component of the National Science Foundation's broader impacts criterion. *Social Epistemology*, *23*(3-4), 249-266.

Joseph, D. (2004). The practice of design-based research: Uncovering the interplay between design, research, and the real-world context. *Educational Psychologist*, *39*(4), 235-242. doi:10.1207/s15326985ep3904_5

Kafai, Y. B., & Burke, Q. (2013, March). The social turn in K-12 programming: Moving from computational thinking to computational participation. In *Proceeding of the 44th ACM technical symposium on computer science education*, 603-608.

Kernighan, B. W., & Ritchie, D. M. (1988). *The C programming language* (2nd ed.). Upper Saddle River, NJ: Prentice Hall PTR.

Koch, M., Lundh, P., & Harris, C. J. (2019). Investigating STEM support and persistence among urban teenage African American and Latina girls across settings. *Urban Education*, *54*(2), 243–273. https://doi.org/10.1177/0042085915618708

Lachney, M. (2017). Culturally responsive computing as brokerage: toward asset building with education-based social movements. *Learning, media and technology*, *42*(4), 420-439.

Ladson-Billings, G. (2014). Culturally relevant pedagogy 2.0: A.K.A. the remix. *Harvard Educational Review*, *84*(1), 74-84.

Lebow, D. (1993). Constructivist values for instructional systems design: Five principles toward a new mindset. *Educational Technology Research & Development*, *41*(3), 4-16.

MacPhee, D., Farro, S., & Canetto, S. S. (2013). Academic self-efficacy and performance of underrepresented STEM majors: Gender, ethnic, and social class patterns. *Analyses of Social Issues & Public Policy*, *13*(1), 347-369. https://doi:10.1111/asap.12033

Maloney, J., Peppler, K., Kafai, Y. B., Resnick, M., and Rusk, N. (2008). *Programming by choice: Urban youth learning programming with Scratch.* Paper presented at 39th SIGCSE Technical Symposium on Computer Science Education, Portland, OR (pp. 367-371). New York: ACM. Retrieved from https://files.eric.ed.gov/fulltext/ED521157.pdf

Meador, A. (2018). Examining recruitment and retention factors for minority STEM majors through a stereotype threat lens. *School Science & Mathematics*, *118*(1/2), 61-69. https://doi:10.1111/ssm.12260

Merriam, S. B., Caffarella, R. S., & Baugartner, L. M. (2007). *Learning in adulthood: A comprehensive guide* (3rd ed.). San Francisco, CA: Jossey-Bass.

McGee, E. (2015, May 1). Why black students struggle in STEM Subjects: Low expectations. *The New Republic*. Retrieved from https://newrepublic.com/article/121693/why-black-males-struggle-stem-subjects

Minium, E. W., Clarke, R., & Coladarci, T. (1999). *Elements of statistical reasoning* (2nd ed.). Hoboken, NJ: John Wiley & Sons, Inc.

Melguizo, T., & Wolniak, G. C. (2011). The earnings benefits of majoring in STEM fields among high achieving minority students. *Research in Higher Education*, *53*(4), 383-405.

MIT App Inventor (n. d.). About us. Retrieved from http://appinventor.mit.edu/about-us

Morgan, G. A., Gilner, J. A., & Harmon, R. J. (2000). Internal validity. *Journal of the American Academy of Child and Adolescent Psychiatry*, *39*(4), 529-531.

Moursund, D. G. (2005). *Introduction to Information and Communication Technology in Education*. Retrieved from https://darkwing.uoregon.edu/~moursund/Books/ICT/ICTBook.pdf

Mouza, C., Marzocchi, A., Pan, Y. C., & Pollock, L. (2016). Development, implementation, and outcomes of an equitable computer science after-school program: Findings from middle-school students. *Journal of Research on Technology in Education*, *48*(2), 84-104.

Nakagawa, S. (2004). A farewell to Bonferroni: The problems of low statistical power and publication bias. *Behavioral Ecology*, *15*(6), 1044-1045.

National Academy of Sciences, National Academy of Engineering, and Institute of Medicine. (2011). *Expanding underrepresented minority participation: America's science and technology talent at the crossroads*. Washington, DC: National Academies Press. Retrieved from https://www.nap.edu/read/12984/chapter/1

National Assessment of Educational Progress. (2015). *The nation's report card*. Retrieved from https://www.nationsreportcard.gov/

National Center for Education Statistics. (2018). Digest of education statistics, 2016 (NCES 2017-094). Retrieved from https://nces.ed.gov/programs/digest/d17/tables/dt17_322.30.asp?current=yes

National Science Board. (2018, January). Science and engineering indicators 2018 (NSB-2018-1). Retrieved from https://nsf.gov/statistics/2018/nsb20181/downloads3

National Science Foundation. (2017-a). Table 2-8: Intentions of freshmen to major in S&E fields, by race or ethnicity and sex: 2014. In *Women, minorities, and persons with disabilities in science and engineering* (Special report NSF 17-310). Arlington, VA. Retrieved from https://www.nsf.gov/statistics/2017/nsf17310/static/data/tab2-8.pdf

National Science Foundation. (2017-b). Table 5-3: Bachelor's degrees awarded, by field, citizenship, ethnicity, and race: 2004–14. In *Women, minorities, and persons with disabilities in science and engineering* (Special report NSF 17-310). Arlington, VA. Retrieved from https://www.nsf.gov/statistics/2017/nsf17310/static/data/tab5-3.pdf

National Science Foundation. (2017-c). Table 9-4. Employed persons 16 years and older, by detailed occupation, ethnicity, and race: 2006–15. In *Women, minorities, and persons with disabilities in science and engineering* (Special report NSF 17-310). Arlington, VA. Retrieved from https://www.nsf.gov/statistics/2017/nsf17310/static/data/tab9-4.pdf

National Science Foundation. (2017-d). Table 9-19. Employed scientists and engineers, by sector of employment, broad occupation, sex, ethnicity, race, and disability status: 2015. In *Women, minorities, and persons with disabilities in science and engineering* (Special report NSF 17-310). Arlington, VA. Retrieved from https://www.nsf.gov/statistics/2017/nsf17310/static/data/tab9-19.pdf

National Science Foundation. (2017, January). Women, minorities, and persons with disabilities in science and engineering: 2017 (Special report NSF 17-310). Arlington, VA. Retrieved from https://www.nsf.gov/statistics/2017/nsf17310/static/downloads/nsf17310-digest.pdf

Noonan, R. (2017). *STEM jobs: 2017 update (Report No. 02-17)*. Washington, DC: The Economics and Statistics Administration. Retrieved from https://www.commerce.gov/sites/default/files/migrated/reports/stem-jobs-2017-update.pdf

Olszewski-Kubilius, P., Steenbergen-Hu, S., Thomson, D., & Rosen, R. (2016). Minority achievement gaps in STEM: Findings of a longitudinal study of Project Excite. *Gifted Child Quarterly*, *61*(1), 20-39.

Ormrod, J. E. (2008). *Human learning* (5th ed.). Upper Saddle River, NJ: Pearson Prentice Hall.

Palmer, R. T., Maramba, D. C., & Daney, Jr., T. E. (2011). A qualitative investigation of factors promoting the retention and persistence of students of color in STEM. *The Journal of Negro Education*, *80*(4), 491-504.

Papadakis, S., Kalogiannakis, M., Orfanakis, V., & Zaranis, N. (2014, June). Novice programming environments. Scratch & App Inventor: A first comparison. In *Proceedings of the 2014 workshop on interaction design in educational environments*, Albacete, Spain (pp. 1-7). New York, NY: ACM. Retrieved from https://www.researchgate.net/publication/266613872_Novice_Programming_Environments_Scratch_App_Inventor_a_first_comparison

Pizarro, J., Guerrero, E., & Galindo, P. L. (2002). Multiple comparison procedures applied to model selection. *Neurocomputing*, *48*(1-4), 155-173.

Price, J. (2010). The effect of instructor race and gender on student persistence in STEM fields. *Economics of Education Review*, *29*(6), 901-910.

Price, P. C., Jhangiani, R., & Chiang, I. A. (2015). *Research methods in psychology* (2nd ed.). Retrieved from: https://opentextbc.ca/researchmethods/

Rahman, F. (2018). *From App Inventor to Java: Introducing object-oriented programming to middle school students through experiential learning*. Paper presented at 2018 ASEE Annual Conference & Exposition, Salt Lake City, Utah. Retrieved from https://peer.asee.org/30539

Repenning, A., Webb, D., & Ioannidou, A. (2010, March). Scalable game design and the development of a checklist for getting computational thinking into public schools. In *Proceedings of the 41st ACM technical symposium on computer science education* (pp. 265-269). ACM.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM, 52*(11), 60-67.

Roberts, T., Jackson, C., Mohr-Schroeder, M. J., Bush, S. B., Maiorca, C., Cavalcanti, M., Schroeder, D. C., Delaney, A., Putnam, L., & Cremeans, C. (2018). Students' perceptions of STEM learning after participating in a summer informal learning experience. *International Journal of STEM Education, 5*(1), 1-14. http://dx.doi.org/10.1186/s40594-018-0133-4

Ryoo, J. J., Margolis, J., Lee, C. H., Sandoval, C. D., & Goode, J. (2013). Democratizing computer science knowledge: Transforming the face of computer science through public high school education. *Learning, Media and Technology, 38*(2), 161-181.

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

Sahin, A. (2013). STEM clubs and science fair competitions: Effects on post-secondary matriculation. *Journal of STEM Education: Innovations and Research*, *14*(1), 5-11.

Scott, K. A., Sheridan, K. M., & Clark, K. (2015). Culturally responsive computing: A theory revisited. *Learning, Media and Technology*, *40*(4), 412-436.

Shadish, W., Cook, T. D., & Campbell, D. T. (2002). *Experimental and quasi-experimental designs for generalized causal inference*. Boston, MA: Houghton Mifflin.

Shackelford, B., & Jankowski, J. (2016, April). Information and communications technology industries account for $133 billion of business R&D performance in the United States in 2013. Retrieved from https://www.nsf.gov/statistics/2016/nsf16309/nsf16309.pdf

Sherman, M., Martin, F., Baldwin, L., & DeFilippo, J. (2014, Sept. 1). App Inventor project rubric—computational thinking through mobile computing. Retrieved from https://nsfmobilect.files.wordpress.com/2014/09/mobile-ct-rubric-for-app-inventor-2014-09-01.pdf

Shingala, M. C., & Rajyaguru, A. (2015). Comparison of post hoc tests for unequal variance. *International Journal of New Technologies in Science and Engineering*, *2*(5), 22-33.

Song, L., & Hill, J. R. (2007). A conceptual model for understanding self-directed learning in online environments. *Journal of Interactive Online Learning*, *6*(1), 27-42.

Soper, D. (n.d.). Critical F-value calculator. *Free Statistics Calculator*. Retrieved from https://www.danielsoper.com/statcalc/calculator.aspx?id=4

Strayhorn, T. L. (2015). Factors influencing black males' preparation for college and success in STEM majors: A mixed methods study. *Western Journal of Black Studies*, *39*(1), 45-63.

Teddlie, C., & Yu, F. (2007). Mixed methods sampling: A typology with examples. *Journal of Mixed Methods Research*, *1*(1), 77-100.

Tsai, C.-Y. (2019). Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy. *Computers in Human Behavior*, *95*, 224–232.

Tyler-Wood, T., Knezek, G., & Christensen, R. (2010). Instruments for assessing interest in STEM content and careers. *Journal of Technology and Teacher Education*, *18*(2), 345-368.

United States Census Bureau. (n.d.-a). Decennial census of population and housing. Retrieved from https://www.census.gov/programs-surveys/decennial-census/decade.2010.html

United States Census Bureau. (n.d.-b). QuickFacts United States. Retrieved from https://www.census.gov/quickfacts/fact/table/US/PST045218

United States Census Bureau (2014). Census Bureau reports majority of STEM college graduates do not work in STEM occupations (Report No. CB14-130). Washington, DC: United States Census Bureau. Retrieved from https://www.census.gov/newsroom/press-releases/2014/cb14-130.html

Vanderhoven, E., Schellens, T., Vanderlinde, R., & Valcke, M. (2016). Developing educational materials about risks on social network sites: a design based research approach. *Educational Technology Research & Development*, *64*(3), 459-480.

Vrasidas, C., Zembylas, M., & Glass, G. V. (2009). *ICT for Education, Development, & Social Justice*. Charlotte, NC: Information Age Publishing, Inc.

Wang, F., & Hannafin, M. J. (2005). Design-based research and technology-enhanced learning environments. *Educational Technology Research and Development*, *53*(4), 5-23.

Wang, X. (2013). Why students choose STEM majors: Motivation, high school learning, and postsecondary context of support. *American Educational Research Journal*, *50*(5), 1081-1121.

Watson, K., & Froyd, J. (2007). Diversifying the US engineering workforce: A new model. *Journal of Engineering Education*, *96*(1), 19-32.

Williams, L. J., & Abdi, H. (2010). Fisher's least significant difference (LSD) test. Encyclopedia of research design, 218, 1-6. Retrieved from http://www.utd.edu/~herve/abdi-LSD2010-pretty.pdf

Won, S. G. L., Evans, M. A., Care, C., Schnittka, C. G. (2015). Youth appropriation of social media for collaborative and facilitated design-based learning. *Computers in Human Behavior*, 50, 385-391,

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends*, *60*(6), 565-568.

Zarrett, N. R., & Malanchuk, O. (2005). Who's computing? Gender and race differences in young adults' decisions to pursue an information technology career. *New Directions for Child and Adolescent Development*, *2005*(110), 65-84.

APPENDICES

Appendix A

Rubric for grading apps

	1	2	3	4	
1. Screen Interface	Single screen with five or fewer visual components that do not programmatically change state.	Single screen with more than five visual components that do not programmatically change state.	Single screen, where some components programmatically change state based on user interaction with the app.	Two or more screens; screens may be implemented as screen components, or by programmatically changing visibility of groups of visual components.	*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness of UI implementation:* [ ] broken (e.g.; buttons not connected to code; interface crashes in some cases) [ ] almost works (minor UI problems) [ ] correct
2. Naming: Components Variables Procedures	Few or no names were changed from their defaults.	Approximately half of names have been changed from their defaults.	Nearly all of names have been changed from their defaults.		*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Names are descriptive of functionality:* [ ] no [ ] yes

3. Events	Fewer than two types of event handlers. (Multiple buttons, all with "buttonX.onClick", are of the same type.)	Two or more types of event handlers. If an event handler modifies label state or sprite position, it's still in this category.	One event handler modifies state in a way that will change the opportunity for other event handlers to begin ("interacting event handlers"). Such as: enabling a clock; hiding or revealing a sprite.		*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct
4. Procedural Abstraction	There are no procedures.	There is exactly one procedure, and it is called.	There is more than one procedure: either for code organization (naming chunks of code that are only called once), or code re-use (subroutines used in multiple places), but not both.	There are both procedures for code organization and code re-use.	*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct
5. Globals with Variables or Text Labels	No data abstraction with globals. Values are hard-coded.	Variables and/or text labels provide names to data, whose values may change as the app runs, or may not (equivalent to a named constant).			*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct

6. Component Abstraction	App does not use component abstraction.	App modifies or reads properties of components out of a list using an “any component” block.			*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct
7. Loops	No use of while block, for-each block, or for-range block	Simple loop, using a constant-value control values.	Loop is governed by data that may change, dynamic.	Loop uses control values that connect data together, across multiple lists, or addressing multiple structures.	*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct
8. Conditionals	No conditionals.	Conditionals use comparison of a variable value to a constant value.	Conditionals use comparison of two variable values.		*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct

9. Lists	No lists.	One single-dimensional list.	More than one independent, single-dimensional lists.	A list of tuples, or (equivalently) multiple corresponding lists (a “multi-dimensional list”).	*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct
10. Data Persistence	Data are only stored in variables or UI component properties, and do not persist when app is closed.	Data persist beyond a single session of the app. (look for: tinydb, tinywebdb)			*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct
11. Data Sharing	No data sharing.	Shared data are limited to a single piece of information (such as a high score or address).	Shared data are compound structures. (such as a high score with name)	Multiple users of the app read and write the same shared pool of data.	*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct

12. Public Web Services	No web services.	Reads data directly from online data source.	Reads and writes online data source.		*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct
13. Accelerometer & Orientation Sensors	No sensors used.	Accelerometer Shake gesture used to trigger events.	App makes decisions based on sensor data (e.g. controls a sprite).		*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct
14. Location Awareness	No location used.	Accesses location and immediately passes it to built-in features (such as maps)	Accesses location and stores it for later retrieval and use.	Inspects location data numerically, processes this data as a feature.	*Appropriateness:* [ ] too simple [ ] just right [ ] too complex *Correctness:* [ ] broken [ ] almost works [ ] correct

Appendix B

Codebook for Qualitative Data

Affective Domain/Agency

- Cognitive Domain – Shows demonstrates natural ability or rapid gain in ability/knowledge; any expression of understanding; different from the student stating confidence
- Confidence/self-efficacy – Student expresses confidence in some way or states that he/she is self-assured
- Confusion - Any significant statement of confusion by the students. Not just asking for help on a step.
- Motivation – Instances where students that seem excited or reluctant to do something**;** the causes for motivation and the detractions
- Personal relevance – Student states that a topic is of personal interest

Barriers and Conflict

- Attrition – any reason why someone doesn't attend, quits, leaves early, or arrives late
- Barriers – system failures, schedule conflicts, anything that makes AMAYS difficult to attend or proceed
- Complaints – any criticism, valid or not
- Conflict – any arguments

Instructional Design

- Coins – References to coin collection or app uploading
- Incentives – any rewards given for completing a task (like prizes)
- Scaffolding – Examples where the support is pushing students to figure it out more on their own
- Tech support – Any support given to resolving tech issues (logging in, using software, using hardware failure). Major issues, not just helping them upload apps.

Social Learning

- Competition – anytime anyone discusses being competitive or competes with peers
- Peer collaboration – students working together directly or concurrently on the same project though at different computers and giving each other help and/or feedback
- Socializing – students that are engaging with each other in ways that are off topic. Conversation not directly relevant to AMAYS
- Mentor - Any information that stands out about the mentor (not the researcher or teacher)
- Teacher support – when the teacher interacts with students in any way; any information that stands out about the teacher
- Monitoring and regulation – Where the support is more disciplinary or getting students back on task. Students attempting to break rules. Surveillance.

Socio-Cultural Norms

- Culture – anything related to heritage, community, economic wealth, audience analysis, context; student identifying as a programmer; anything positive about one's culture/attitude in relation to STEM
- Social and cultural stigmas - (e.g., "programing is for nerds"); concerns regarding representation; anything negative about one's culture/attitude in relation to STEM

Student Activity

- Designing – Planning out an app design, including a custom app/DIY (before working on it); storyboarding, additional visual/media components, etc.
- Inactivity – when a student is choosing to not work on AMAYS projects or cant due to any system failure; don't need to code with motivation
- Remix, Custom, Independent work - Major changes to an app that are in the process of being made or changes that have been completed. Independent work. Any creative expression beyond what is in the packets.
- Sharable assets – Examples of students showing off their apps or wanting to show them off (on an App Store, to each other or to an adult in class, etc.)
- Task Completion – Something is completed

Appendix C

The most relevant questions from the student interviews (rephrased to hide identifying elements)

- What made you want to sign up for the after-school intervention?
- How much did you like what you were doing?
- What was your favorite part?
- What was your least favorite part?
- Were there any times when you felt like you were good at what you were doing? If so, can you describe those times?
- What kind of things affected your ability to come to the intervention? What would help you or make you want to come to the intervention more often?
- How did making real apps that other people can download and use make you feel?
- Did you work with another student at any point?
- Would you like to continue with activities like these? Will you create apps on your own outside of the program? Why or why not?
- What were the most important or coolest things you learned from your time in the intervention?
- How would you make these activities more fun?