

Creating Socially Relevant Mobile Apps: Infusing Computing into Middle School Curricula in Two School Districts

Lijun Ni, University at Albany, lni@albany.edu
Fred Martin, University of Massachusetts Lowell, fredm@cs.uml.edu

Abstract: In this paper, we share our experiences implementing a professional development program in two school districts with middle school teachers who integrated an introductory computer science curriculum into their teaching. The 15 to 20-hour curriculum was based on students collaboratively creating mobile apps for socially relevant purposes with MIT App Inventor. Eleven teachers infused the curriculum into technology, math, engineering, library and art courses. We investigated how teachers modified the curriculum to fit their respective standards and students' needs. We discuss the challenges they faced and propose ways of addressing these barriers. We found that the teachers were successful in combining digital literacy skills with computer science—not only to facilitate students' learning, but also to connect with their diverse ethnic backgrounds and their contemporary passions.

Introduction

This study is part of a larger project, Middle School Pathways in Computer Science (CS Pathways), aimed at establishing a sustainable, institutionalized computer science curriculum at the middle school level in two school districts. Working with the districts' existing teachers, the project is infusing computer science into the districts' existing curriculum, with an explicit focus on creating mobile apps that address local community needs. When fully implemented, the project will provide an introductory computer science experience to all middle school students in these two districts.

The funding program supporting the project is named Innovative Technology Experiences for Students and Teachers (ITEST). This program “promotes PreK-12 student interests and capacities to participate in the science, technology, engineering, and mathematics (STEM) and information and communications technology (ICT) workforce of the future”(NSF, 2016a). While the program is focused on student interest and pursuit of STEM careers, it explicitly recognizes the crucial role of teachers in this process. Our project, and this study, focus on this research question: “What instructional and curricular models can effectively engage teachers to utilize and integrate technologies so as to enhance student understanding of STEM-related occupations?” (NSF, 2016b). This project demonstrates the challenges and opportunities in working with teachers who have a range of backgrounds, including technology, library, engineering, and mathematics, understanding what is necessary for them to feel supported and be effective in teaching computer science to all students.

Background

There is much evidence showing a need for greater high-tech skills in today's workforce (e.g., Olson & Riordan, 2012). Critically, there is substantial under-representation by women and ethnic minorities in technical fields, including computer science (Jackson, Starobin, & Laanan, 2013). This is a matter of social justice and international competitiveness (Leggon et al., 2015). Addressing this, since 1999 NSF has spearheaded a series of funding programs to “broaden participation in computing” and other STEM fields (Aspray, 2016). Most recently, the White House announced Computer Science For All, which strives to “empower a generation of American students with the computer science skills they need to thrive in a digital economy” (Smith, 2016).

In order to reach all students, it is necessary to have a curriculum and pedagogical approach that will engage all students. The Exploring Computer Science (ECS) project has demonstrated how to bring values of equity and inclusiveness into the classroom (Goode, Chapman, & Margolis, 2012). Furthermore, research has shown the power of engaging students from underrepresented groups with culturally relevant examples of computing (e.g., Eglash's culturally situated design tools) and a clear social purpose to computing with real-world applications (Buckley, Nordlinger, Subramanian, 2008; Eglash, Gilbert, & Foster, 2013; Fisher, & Margolis, 2002). The CS Pathways project's focus is to engage students in learning computing through creating mobile apps for socially beneficial purposes.

We use MIT App Inventor, a blocks-based programming system, as the design environment for students to create mobile apps that address local community needs. In App Inventor, students can simply drag and drop blocks of code to create an app, which can be downloaded to an Android mobile phone or tablet. Prior work has demonstrated its success in providing positive computing experience to students with its expressive power (Wagner et al., 2013; Sherman & Martin, 2015; Ni et al., 2016). We believe that App Inventor can be

used to democratize computing—to provide the expressive power of computing to all learners, not only the “small group of technically elite” (Wolber, et al., 2015). The CS Pathways computing curriculum was based on students collaboratively creating apps for socially relevant purposes with App Inventor.

Partnership and teachers

The project partnership is with the school districts of Medford and Everett in MA. Medford has 63% white, 15% African-American, and 10% Hispanic students with 44% high needs students. This is approximately equivalent to state averages. Everett has 31% white, 18% African-American, and 44% Hispanic students with 62% high needs students. This is more diverse and low-social economic status than state averages. The districts have a history of collaborating on technology initiatives via a non-profit that was created with the districts’ support. Technology is a required subject at the middle school level in both districts; by infusing computer science into this course, all students would get an introduction to computer science. In the project design, all of the districts’ middle school technology teachers would participate. Other teachers were invited to participate too.

In the project’s first year, we recruited Cohort 1 including: one of the two technology teachers, an engineering teacher, and an art teacher in Medford; two of the five technology teachers in Everett. In the second year, Cohort 2 included the other technology teachers from both districts, a librarian, and a math teacher. In its final year, Cohort 3 will consist of a replacement technology teacher and a science teacher (see Figure 1).

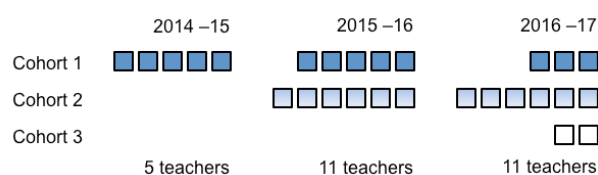


Figure 1. Teacher Cohorts

Professional development

The professional development (PD) was designed as an ongoing, multi-year process to introduce teachers to computer science content and pedagogy, support them in classroom implementation, and encourage them to share their learning with each other. The PD was led by the Teacher Learning Center Director, a full-time staff member hired for the project, and the second author, with contributions from the whole project team.

PD goals and structure

The professional development encompassed 38 hours of instructional, meeting, and homework time per teacher per year. Our project was funded to begin in Fall 2014; thus a summer PD with the first cohort of teachers was not possible. Instead, we organized a series of ten 2-hour afterschool meetings, beginning in October 2014 and ending in January 2015. For the second cohort, we ran a one-week summer camp. In both cases, for a given teacher’s first year, the content of the PD was the same, and included these elements:

1. Introduction to building apps in MIT App Inventor, including use of digital media (images, sound) and blocks-based programming;
2. Computer Science pedagogy of equity and inclusiveness (e.g. contesting the “geek gene”, using pair programming, understanding the values of the ECS project);
3. Engaging teachers in an original app design process, from brainstorm to planning to completion, so they could facilitate this with their students;
4. Lesson planning, including integration of new material and addressing standards.

For the Cohort 1 teachers in their second year, we included differentiated PD. We conducted a series of four mini-workshops. Each included readings, a homework assignment, and debrief conversation that was conducted via video conference. The monthly topics included assessment of student work, pedagogical content knowledge as it applies to computer science, social impact of computing including career opportunities, and advanced programming in App Inventor (e.g., using index variables and lists).

Methods

Our project’s external evaluator administrated baseline surveys and end-of year surveys with the 11 teachers to assess the quality of the PD program and teachers’ experience and attitudes. The Teacher Learning Center Director regularly visited project classrooms and took observation notes. The last PD session of the school year

focused on group reflection on how the curriculum was implemented. The project researchers took meeting notes and discussed afterward. This informed the development of a teacher interview protocol.

We use interviewing as our qualitative method to explore teachers' curriculum implementation experiences (Seidman, 2005). Interviews were conducted to further understand the variation in their curriculum design and implementation, and barriers or opportunities for implementation in different classroom contexts. Teachers were asked to describe how they used/modified the model lesson plans, to what extent the curriculum was implemented, what challenges they encountered, and the most helpful things supporting their teaching. We interviewed eight of the eleven teachers in Spring 2016, representing six schools and both school districts.

The next section presents the stories teachers reported about their curriculum implementation. These teacher reflections have been triangulated with our analysis of group reflection, classroom observations, and interviews.

Findings

Teachers gave high ratings to the professional development. Most teachers reported that the overall quality of the PD, app development support, and overall usefulness as "good" (44%) or "excellent" (44%). Teachers also reported increased confidence in using apps, creating apps, and using computer terms ($p < .05$). In addition, Cohort 2 teachers reported an increased frequency of creating apps ($p < .05$).

During the past school year, all the 11 teachers implemented the CS Pathways curriculum into existing courses, reaching an aggregate total of over 750 students. Next sections report stories from the eight teachers participated in our research interviews.

Curriculum implementation

The eight teachers we interviewed include three teachers from Cohort 1 and five teachers from Cohort 2. The five teachers (P1-P5) came from one district and P6-P8 were from the other district. These teachers taught the CS Pathways curriculum within technology, engineering, library, or mathematics classes with 6th, 7th or 8th graders (see Table 1). The curriculum was implemented with different class schedules. Teachers from one school district met students more often while the other teachers saw students once every 6 days, 8 days or 9 days. The research team also investigated the project's impact on students' attitudes and computational thinking skills through surveys and analysis of student work. The focus of this paper is on teacher PD and curriculum implementation, and results on student learning are being prepared for separate publication.

Table 1: Interviewed Teacher Information

Teacher	Cohort	School	Area/Course	Schedule	Grade
P1	1	A	Technology	6-day cycle	7 th 8 th
P2	1	B	Technology	8-day cycle	8 th
P3	2	C	Library	8-day cycle	6 th
P4	2	C	Technology	9-day cycle	8 th
P5	2	D	Technology	8-day cycle	7 th 8 th
P6	1	E	Engineering	Twice/week	6 th
P7	2	F	Math	3 times/week	6 th
P8	2	E	Technology	Twice/week	8 th

Model lesson plans

During the 1st year of the project, Cohort 1 teachers developed and implemented a 15–20 hour computing curriculum with App Inventor. Prior to the start of the project's 2nd year, a series of model lesson plans were created from the teachers' work and shared with all the project teachers. These lesson plans were drawn from the work of three Cohort 1 teachers. Two were from the same school district and had collaborated closely on creating each lesson. The third teacher's lesson plans used a faster pace with varied starter apps. The Teacher Learning Center Director edited the curriculum materials from these three teachers by linking CSTA curriculum standards (Seehorn et. al, 2011), checking overall sequence, and using a standardized lesson plan template.

Students were introduced to App Inventor and then had three to five classes to create original final apps. They had opportunities to share their work to the class or the whole school community. The curriculum included one lesson introducing students to computing and career pathways, and another lesson inviting local computing professionals to visit the classrooms. Table 2 shows the overall curriculum sequence.

The model lesson plans provided a map for Cohort 2. All teachers were encouraged to modify the lessons based on their own classroom needs. They infused the lessons into their existing curricula with a variety

of strategies, including using different starter apps to motivate and inspire students, integrating digital literacy, and changing sequence.

Table 2: Model Curriculum Sequence

Day	Overview of Lessons	Day	Overview of Lessons
1	Pre-survey; using your Google login	9	Media app with if-then-else programming
2	Overview of computing and career fields	10	Apps for social good / final project brainstorming
3	Code.org block programming intro / CS Unplugged	11	Storyboarding final project app
4	Introduction to tablet and App Inventor	12	Working on final project app
5	First app: Talk To Me (text to speech)	13	Working on final project app
6	Talk To Me part 2 and sharing	14	Finish final app; export source / installer files
7	Media player app using w/preloaded starter code	15	Computing professional visit
8	Digital literacy: gathering own media for apps	16	Post-survey and app showcase

Adapting the model lesson plans

Integrating with existing curriculum

The project teachers integrated the CS Pathways computing curriculum into technology/computer, engineering, library, math, and art classes with students from grade 6 to 8. All the eight interviewed teachers reported how they felt this computing curriculum could fit into the existing middle school curricula.

Computer/Technology Course

The technology teachers recognized that the curriculum included digital literacy components, which were consistent with their existing technology curriculum. For example, students used Google search to find pictures to make slides. They searched pictures and presented them in apps too. A few teachers explicitly added relevant digital literacy content into the lesson plans: e.g., how to download and upload images and sounds to apps. In addition to seeing the overlapped content of digital literacy, the technology teachers also understood that the CS Pathways curriculum taught more computer science beyond applications. They saw the values of teaching computational thinking within the technology curriculum as “high-level things”. One technology teacher said:

“You have an idea, then you start thinking, that’s the computational thinking. I introduced to them you have to solve a problem; you have to breakdown each steps, you have to use blocks... These are the higher level things.”

Engineering Course

The engineering teacher felt that it was challenging to cover the required engineering content with 15–16 hours focused on computer science of the 40 hours total of class time. As a compromise, he experimented with an integration approach by asking students to develop final apps around one of the six engineering topics in this course. Students developed engineering-focused apps, such as informational apps on transportation, machine tools or game apps on futurist flights. The engineering teacher explained how this change worked for his class:

“I started with social good apps for communities, but there is an issue for me. If I have 15–16 hours as computer science, that’s a huge chunk of class time. So we came to a compromise: the final apps have to do with engineering. They spent 3–5 classes on the final apps. They were thinking about engineering and focusing on engineering. It did make me feel a lot better because obviously that’s part of my curriculum... I want them to see what kind of apps they could make and give them some tools, and then they can make their own engineering apps.”

Library Course

The library teacher was excited about finding ways of integrating the computing curriculum into a library course. She changed the media player sample lesson, which was based on showcasing Martin Luther King’s speeches, to bibliography apps on book authors. She identified relevant curriculum components in library curriculum standards. She reported this as a good opportunity of integrating computing into the library course.

“Here is all about the author, title... the students can choose a book to make a game, or informational app, some type of interactive app... I found the American Association for School Library has a framework that is more technology-oriented. The copy of that is almost

of the CSTA standards, and there was some overlap, something like digital citizenship; there were also things like acceptable use of technology... I was able to tie with a lot of that.”

For students’ final apps, the teacher relaxed the constraint of book authors only. She allowed a student to create a bibliography app about pop star Justin Bieber. The student recorded and uploaded his own voice about Bieber’s biography and using the media player component to play the audio. We support this approach of connecting to students’ interests.

Mathematics Course

The math teacher saw the value of teaching logical thinking through this curriculum and felt it fit well with mathematics standards. He adopted the starter lesson on “if-then-else,” which was the media player using content from Martin Luther King’ speeches. Students were provided with a starter app and asked to change the app to control the media player to play and pause a media file. The math teacher explained his choice as below:

“I did the if-then-else. I think it’s very useful structure. I try to teach programming, following sequences. This is very good basic logic structure that particularly adds its lens to the implementation. I probably focus more on this than other teachers, with a goal of building computational thinking. It is perfect for the math—how to model, structure... they need to structure their thinking in order to do the apps as they want to do. There is a huge need for the students to develop to be successful.”

This teacher also relaxed the content requirement and allowed students to choose material based on their own interests. For him, this gave the students motivation to engage with the important if-then-else logic.

Customizing starter apps and examples

Teachers used starter apps to scaffold learning. They modified or created new starter apps to meet a variety of students’ interests. The CS Pathways curriculum was implemented with a diverse group of students, including English Language Learner (ELL) students and students with special needs. One teacher had a group of ELL students creating educational apps to share their cultural heritage. A group of students created a whimsical “Cheese Around The World” app. Each of five students recorded themselves saying the word for cheese in the language they spoke at home. They incorporated these into an app with flags from each country. Clicking on a flag will make the app say “cheese” in that country’s language. This was an example of modification that supported a culturally diverse group of students’ self-expression and creativity—a simple but still powerful version of Eglash’s culturally situated computing. Another teacher had her special-needs students make a zoo menagerie—instead of the media player app based on the Martin Luther King’s speeches. This teacher assigned each of her students a different animal. Each student located a photo and sound of their animal, and made a player app from those digital media components. This teacher had previously shown her students the basic HelloPurr app, which was much less complicated.

Centered on the theme of social-good/community-needs, another teacher created and presented a school introduction app to her students. She recorded her own voice introducing the school and welcoming students and families as the principle with a map of the school buildings included in the app. Students felt this app was very cool and got inspired and motivated to learn about creating apps, including how to record audios, upload media, using media player component and buttons.

Integrating digital literacy with computer science

Through the first year of PD program, we identified the need of teaching digital literacy to our participating teachers while they were learning computer science content. Over the process of implementing the CS Pathways curriculum, teachers also found that students were not well equipped with digital literacy skills. They needed to teach relevant digital literacy to support students’ learning of computing concepts, especially with 6th graders, who had not learned most of the computer literacy involved in this curriculum. In particular, the App Inventor curriculum requires using Gmail accounts and working on media resources. Teachers identified the need of explicitly teaching these skills. While following with the model lesson plans, teachers integrated relevant digital literacy content within the lessons. This work, which arose naturally from projects involving App Inventor, dovetailed with both the technology teachers’ existing curriculum standards and Massachusetts’ new Digital Literacy and Computer Science K-12 standards (Chester, 2016).

Facilitating implementation

Teachers reported three aspects of the projects were most helpful for facilitating their teaching of the new curriculum: the model lesson plans, regular PD meetings, and ongoing support from the project team.

The impact of PD

Teachers reported that the PD sessions were very helpful. They especially saw the value of being able to communicate with other teachers regularly for sharing, reflecting on their teaching and supporting each other. As one teacher said:

“It’s good for the teachers to collaborate together. I like to meet face to face because there is always someone in the group comes up and offer something for the conversion. Say, that sounds a great idea. I have got some ideas from [teacher name]. He has lots of great ideas... And someone might say, I haven’t thought about that. I want to try that too... Especially, when you first start, it’s nice to see others have maybe intimidation, make you feel a little better. The roundtable sessions are really instrumental. It offers a good start.”

Ongoing support for teachers

This project offered ongoing support for teachers in implementing the CS Pathways curriculum. The Cohort 1 teachers participated in another four online mini-PD sessions. Cohort 2 teachers received regular classroom visits from the Teacher Learning Center Director, as well as computer science undergraduate and graduate students to help with troubleshooting technical issues and answering curriculum-related questions. Teachers all reported these visits as very “helpful.” Teachers realized the need of ongoing learning, and they also felt comfort with seeking support from the project team and other teachers.

“I need support for some more advanced things... Sometimes students want to make things more game like, maybe things such as a clock timer... I need more practice myself, I would need support in such things, but I know where to find resources. If I feel really stuck, I can email [project team] or anyone involved in the project. They are more than happy to help.”

Model lesson plans and handouts

The three Cohort 1 teachers felt proud of their own contributions to the model lesson plans. Meanwhile, the Cohort 2 teachers used the lesson plans to start their own explorations. Even if they modified the lessons, all teachers appreciated the availabilities of these lessons. One teacher referred to the lessons as a “life saver.” The Teacher Learning Center Director also worked with teachers to create a series of class handouts. These handouts outlined specific apps or introduced technical settings (e.g., how to use the tablets as audio recording devices). Teachers felt these handouts were very helpful in reducing their anxiety and supporting their teaching. One teacher said:

“The model lesson plans are life saver... For me, I really like to have cards on that, for students to use to refresh memory, or just check things where it is that is working in the blocks, to clarify things. And have handouts for them to see... If you rename a button, you may spend 10 minutes to find it.”

Barriers to implementation

Teachers were also asked to reflect on the major challenges and difficulties they encountered in implementing the curriculum. During the interviews, they reported four major challenges: access to labs and technology, the need of ongoing learning, motivating students, and class scheduling.

Access to labs and technology

Teachers reported it was very challenging to access labs and set up technical settings ready for the class. Some lost access to their labs for significant periods of time because the computers were needed to administer standardized tests. Teaching this curriculum required access to devices (computers and tablets), WiFi, and student Gmail accounts to log into App Inventor. Any problem with these technical components could consume class time and prevent learning. For example, one teacher was struggling with an unstable network. Another teacher described another situation she encountered and felt very frustrated:

“The kids were all assigned Gmail accounts and they were broken into pairs... After they got all the introduction done and started implementing easier apps, they started getting better at it... One of our technicians changed the kids’ email accounts. They couldn’t access their original App Inventor accounts where their apps were in. That was very frustrating to my students, which robbed them off enthusiasm. Then my lab was shut down for over a month due to PARCC testing and special-ed testing... It’s hard to bring the kids back to the flow... I did bring them back with new accounts... but I lost lab for 5 weeks, very frustrated.”

The project team sent staff to help with setting up devices, which offered great support for the teachers to start with working devices. This kind of support addressed part of the technical issues. Teachers needed to prepare lessons that would advance their goals in the face of technology failures. Meanwhile, there was still a great need of technical support at the school level for the curriculum implementation.

Need of ongoing learning

Although teachers reported an increase of their self-confidence in teaching the curriculum through post-school year surveys, they felt a strong sense of in need of ongoing learning. Four teachers reported this as a challenge for their teaching. First, the curriculum was new for them, including technology/computer teachers. Teachers felt the curriculum content was new and difficult. One technology teacher said:

“It’s difficult for me. I would like to have more support... I still don’t feel I have known enough to run this all by myself... For me, coding is like learning a new language... It’s difficult. I know definitely I have learned a lot and the kids have learned a lot too.”

Another teacher further explained that she also felt challenged when students started making their own apps. She needed to learn more to be able to help the students.

Social good and student motivation

Two teachers reported another challenge related to motivating students. The project had a major focus on creating apps for social good, which was designed to motivate students’ learning. However, teachers found some of the students were not excited about creating social-good apps. Therefore, they needed to find other ways of motivating students. One teacher allowed her students to create apps on other topics of their interest.

“When they were told they could make their own apps for something social good, to solve a problem, they were not that thrilled about it. They want to make something interesting to them... I want to find some middle ground for them. Making apps is so exciting, but making social good apps are not so to them.... I want them to do something. A lot of them go towards games. I’m ok with that... I’m starting, I keep thinking of different ways of doing things.”

Looking ahead, we will encourage teachers to consider any apps that connect to students’ personal interests as socially relevant, supporting a culturally diverse group of students’ self-expression and creativity.

Scheduling and standards

Teachers reported time/schedule-related issues as the biggest challenge. Teachers from one district had a long-cycle schedule: teachers met students every 6 days, 8 days or 9 days. Ideally, teachers would like to see students more often. Teachers felt this long-cycle schedule made it hard to teach the curriculum. On the other hand, the math teacher and engineering teachers felt the pressure of competing academic requirements together with teaching the new computing curriculum. The math teacher implemented the curriculum at the second half the school year after completing the required math requirements. He saw the challenge of getting students ready for math tests while providing sufficient time for students in exploring and creating final apps. He reduced class time for students to work on final apps. The Engineering teacher had similar worries. He changed the lesson plans to include engineering-related apps as students’ final projects. He was hoping the new state standards on digital literacy and computer science would be included in the engineering curriculum.

“It’s not in my standards... Computer science is just another thing. There is no room to fit with everything. I hope the new state standards become [part of] what I am supposed to teach.”

Discussion and conclusions

In this study, teachers shared their experiences in implementing the computing curriculum with a community focus into their existing curricula. First, their stories have demonstrated how socially relevant computing curriculum could serve a diverse group of students. The students’ “Cheese Around The World” app was a playful way for them to share their cultural heritage with each other. Other apps, such as the Justin Bieber bio app, allowed students to share their passions with each other. We see parallels between Eglash’s culturally responsive computing (Eglash et al., 2013) and our teachers’ support of these kinds of projects. Further, building apps that exercise digital literacy skills is a comfortable on-ramp to projects that involve more complex coding, for teachers and students alike. With Massachusetts’ new standards for digital literacy and computer science (Chester, 2016), we see the opportunities for this approach to become widespread.

This study identifies both facilitators and barriers for teachers’ implementation of this computing curriculum. We see teachers made significant efforts and worked closely with the project team to address a

variety of challenges. Our experiences suggest a few strategies for infusing such a computing curriculum into middle school curricula. First, it can be challenging to work with diverse groups of students with different needs. Teachers can experiment with different strategies for motivating students and scaffolding learning, such as using starter apps, customizing app themes, modifying paces for students with special needs. Second, districts must commit to technology support and access. Teachers are demoralized when their instructional labs are appropriated for computer-based standardized testing. We saw a teacher gaining necessary access to her lab only because the district temporarily reverted to paper-based testing. Third, teaching (new) computing curriculum requires ongoing learning. Teachers value learning, reflection and conversations with their peers.

Our project's immediate goal is to establish an institutionalized computer science curriculum at the middle school level in the two districts. Our 3rd-year PD program is focused on supporting teachers with curriculum revision and institutionalization. Our experiences indicate that it is a long way for teachers to become self-sufficient. It also takes systemic support from the school district and above in terms of administrative support and standards.

References

- Aspray, W. (2016). Opening computing careers to underrepresented groups. *Participation in Computing*. Springer International Publishing, 13-52.
- Buckley, M., Nordlinger, J., Subramanian, D. (2008). Socially relevant computing. *ACM SIGCSE Bulletin*, 40(1), 347-351.
- Chester, M. (2016). Digital Literacy and Computer Science (DLCS) Curriculum Framework. Massachusetts Department of Elementary and Secondary Education.
- Eglash, R., Gilbert, J. E., & Foster, E. (2013). Toward culturally responsive computing education. *Communications of the ACM*, 56(7), 33-36.
- Fisher, A., & Margolis, J. (2002). Unlocking the clubhouse: the Carnegie Mellon experience. *ACM SIGCSE Bulletin*, 34(2), 79-83.
- Goode, J., Chapman, G., & Margolis, J. (2012). Beyond curriculum: The Exploring Computer Science program. *ACM Inroads*, 3(2), 47-53.
- Jackson, D. L., Starobin, S. S., & Laanan, F. S. (2013). The shared experiences: Facilitating successful transfer of women and underrepresented minorities in STEM fields. *New Directions for Higher Education*, 2013(162), 69-76.
- Leggon, C., McNeely, C. L., & Yoon, J. (2015). Advancing women in science: Policies for progress. *Advancing Women in Science*. Springer International Publishing, 307-340.
- Ni, L., Schilder, D., Sherman, M., Martin, F. (2016). Computing with a community focus: Outcomes from an App Inventor summer camp for middle school students. *Journal of Computing Sciences in Colleges*, 31(6), 82-89.
- NSF (2016). Innovative Technology Experiences for Students and Teachers (ITEST) program synopsis. https://www.nsf.gov/funding/pgm_summ.jsp?pims_id=5467. Retrieved August 19, 2016.
- NSF (2016). Innovative Technology Experiences for Students and Teachers program solicitation #15-599.
- Olson, S., & Riordan, D. G. (2012). Engage to excel: Producing one million additional college graduates with degrees in STEM. *Report to the President*. Executive Office of the President.
- Seehorn, D., Carey, S., Fuschetto, B., Lee, I., Moix, D., O'Grady-Cunniff, D., ... & Verno, A. (2011). *CSTA K-12 CS Standards*. <https://csta.acm.org/Curriculum/sub/K12Standards.html>. Retrieved Nov. 4, 2016.
- Seidman, I. (2005). *Interviewing as qualitative research: A guide for researchers in education and the social sciences*. Teachers College Press.
- Sherman, M. & Martin, F. (2015). The assessment of mobile computational thinking. *Journal of Computing Sciences in Colleges*. 30(6), 53-59.
- Smith, M. (2016). Computer Science For All. <https://www.whitehouse.gov/blog/2016/01/30/computer-science-all>. Retrieved November 4, 2016.
- Wagner, A., Gray, J., Corley, J., Wolber, D. (2013). Using App Inventor in a K-12 summer camp. *SIGCSE '13*, 621-626.
- Wolber, D., Abelson, H., Friedman, M. (2015). Democratizing computing with App Inventor. *GetMobile: Mobile Computing and Communications Review*, 18(4), 53-58.

Acknowledgments

We thank our project team members (Akira Kamiya, Molly Laden, Diane Schilder, Farzeen Harunani and Mark Sherman) for their collaboration and all the participating teachers. This material is based upon work supported by the National Science Foundation under Grant No. 1433592.