Travis W. Peters

Ph.D. Candidate, Department of Computer Science, Dartmouth College

traviswp@cs.dartmouth.edu

Research Interests

Computer and Network Security; mHealth and IoT security; Trusted Computing; Mobile and Wireless Systems

Education

Ph.D., Computer Science *2013 - present (expected: June 2019)*
Dartmouth College, Hanover, NH
Thesis Title: Securing Personal Devices and Networks in the Internet of Things (IoT)
Advisor: Dr. David Kotz
Doctoral Committee: Dr. David Kotz, Dr. Sean Smith, Dr. Xia Zhou, Dr. José Camacho

B.S., Mathematics & Computer Science *2008 - 2012*
Western Washington University (WWU), Bellingham, WA

Teaching Experience

Teaching

Problem Solving via Object-Oriented Programming (COSC 10), *Dartmouth College* *Winter 2015*

Teaching Assistantships

Software Design & Implementation (COSC 50), *Dartmouth College* *Spring 2016*
Introduction to Programming & Computing (COSC 1), *Dartmouth College* *Spring 2014*
Problem Solving via Object-Oriented Programming (COSC 10), *Dartmouth College* *Winter 2014*
Introduction to Programming & Computing (COSC 1), *Dartmouth College* *Fall 2013*
Programming Fundamentals in C++ (CSCI 140), *Western Washington University* *Fall 2012*
Teaching Assistant, *Family House Academy* *Summer 2009*

Guest Lectures

Debugging with GDB and Valgrind, *Dartmouth College (COSC 50)* *January 2018, April 2017, April 2016*
A 65-minute lecture on debugging program logic and memory leaks with GDB and Valgrind. This lesson includes an exercise where students get to discover and fix a buffer-overflow vulnerability.
Notes at `https://traviswp.github.io/classes/debugging-gdb-valgrind/`.

Introduction to Pebble Development, *Dartmouth College (COSC 50)* *April 2016*
A 65-minute lecture on programming on Pebble smartwatches and a culminating team project.
Notes at `https://traviswp.github.io/classes/pebble-project-intro/`.

Three Kinds of Memory, *Dartmouth College (COSC 50)* *April 2016*
A 65-minute lecture on understanding the different kinds of memory and basic memory management in C.
Notes at `https://traviswp.github.io/classes/memory/`.

Research Experience

Research Assistant, *Dartmouth College, Hanover, NH* *January 2014 - Present*
Researching security and privacy threats in mobile health (mHealth). We focus on system and network security within personal area networks and body area networks of health and wellness devices. To this end, we design and experimentally validate hardware and software architectures. To learn more, visit our websites: Trustworthy Health and Wellness (`thaw.org`) and The Amulet Project (`amulet-project.org`).

Security Research Intern, *Intel Labs, Hillsboro, OR* *June 2016 - September 2016*
Conducted a survey on security and privacy threats in the Internet of Things (IoT). Presented findings internally and worked with others to develop a larger IoT security research agenda.

Security Research Intern, *Intel Labs, Hillsboro, OR* *June 2015 - September 2015*
Designed a security architecture to enhance Bluetooth security. Our work is built on top of Intel's SGX – state-of-the-art system security features. Our work was published in *HASP'18* and a related patent has been filed.

Industry Experience

DevOps Engineer, *Attachmate, Bellingham/Seattle, WA* *January 2013 - August 2013*

- Designed and built an automated virtual machine (VM) template management infrastructure using Chef and VMware's vCloud Director. The infrastructure automated how VMs running various operating systems (Windows, Red Hat Linux, SUSE) are deployed and maintained (patched & updated).
- Developed automation routines in Ruby, Bash, and Batch (install software, configure machine settings, etc.).
- Wrote and maintained design specifications and unit tests.

Software Engineer Intern, *Attachmate, Bellingham, WA* *August 2012 - December 2012*

- Extended Luminet (enterprise fraud management system) to integrate with various Security Information & Event Management (SIEM) systems. The extensions used our customizable XML configuration file to enable network operators to configure Luminet to log to various SIEMs.
- Demonstrated correctness of code through implementation of unit tests & automated testing methods.
- Presented and demoed project results to the Luminet product team.

Mobile Developer & Intern Team Lead, *Emergency Reporting, Bellingham, WA* *January 2012 - June 2012*

- Designed and implemented a mobile application to aid Fire/Rescue and EMS responders. Our application enabled better in-the-field access to Emergency Reporting's cloud-based record and reporting management system. (Our work spearheaded what is now the InspectER mobile app.)
- Led team of four interns to implement compatible mobile application on iOS and Android platforms.
- Implemented data security (at-rest and in-transit), database access, and integration with Google Maps.

Other Work Experience

Vice President for Business & Operations, *Associated Students of WWU* *June 2011 - June 2012*

- Elected by the student body of Western Washington University (more than 15,000 students).
- Charged with overseeing the internal operations of the Associated Students programs, services, and facilities.
- Managed six other student managers of departments with as many as 20 employees each.
- Facilitated organizational budgeting process, employee hiring process, and internal program assessment.
- Chaired committee to develop operating & non-operating budget for fiscal year 2012 ($3.1 million budget).

Marketing & Technical Associate, *Caso Inc., San Antonio TX* *June 2010 - January 2011*

- Collaborated with the marketing team to implement search engine optimization of company website.
- Advised a team of department leaders to pilot a new organizational management system.

Publications

Travis Peters, Reshma Lal, Srikanth Varadarajan, Pradeep Pappachan, and David Kotz. **BASTION-SGX: Bluetooth and Architectural Support for Trusted I/O on SGX**. In *Proceedings of the Workshop on Hardware and Architectural Support for Security and Privacy (HASP)*, pages 1–9, June 2018.

David Kotz and Travis Peters. **Challenges to ensuring human safety throughout the life-cycle of Smart Environments**. In *Proceedings of the ACM Workshop on the Internet of Safe Things (SafeThings)*, pages 1–7, November 2017.

Josiah Hester, Travis Peters, Tianlong Yun, Ronald Peterson, Joseph Skinner, Bhargav Golla, Kevin Storer, Steven Hearndon, Kevin Freeman, Sarah Lord, Ryan Halter, David Kotz, and Jacob Sorber. **Amulet: An Energy-Efficient, Multi-Application Wearable Platform**. In *Proceedings of the ACM Conference on Embedded Network Sensor Systems (SenSys)*, pages 216–229, November 2016.

Andres Molina-Markham, Ronald Peterson, Joseph Skinner, Tianlong Yun, Bhargav Golla, Kevin Freeman, Travis Peters, Jacob Sorber, Ryan Halter, and David Kotz. **Amulet: A Secure Architecture for mHealth Applications for Low-power Wearable Devices**. In *Proceedings of the Workshop on Mobile Medical Applications - Design and Development (WMMADD)*, pages 16–21. ACM, November 2014.

Travis Peters and Puneet Jain. **MobiSys 2014**. *IEEE Pervasive Computing*, 13(4):93–96, Oct.–Dec. 2014.

Chip Jackson, Lucas Bourne, and Travis Peters. **Computing Along the Big Long River**. *The UMAP Journal for Undergraduate Mathematics & Research*, 33(3):231–246, Fall 2012.

Patents

Srikanth Varadarajan, Reshma Lal, Steven B. McGowan, Hakan Magnus Eriksson, and Travis W. Peters. **System, apparatus and method for providing trusted input/output communications (pending)**, May 2018.

Demos, Posters, Tech Reports, Work in Progress, etc.

Timothy Pierson, Travis Peters, Ronald Peterson, and David Kotz. **Poster: Proximity detection with single-antenna IoT devices (accepted)**. In *Proceedings of the International Conference on Mobile Computing and Networking - Posters (MobiCom'18 Posters)*, August 2018.

Travis Peters. **A Survey of Trustworthy Computing on Mobile & Wearable Systems**. Technical Report TR2017-823, Dartmouth Computer Science, May 2017.

Josiah Hester, Travis Peters, Tianlong Yun, Ronald Peterson, Joseph Skinner, Bhargav Golla, Kevin Storer, Steven Hearndon, Sarah Lord, Ryan Halter, David Kotz, and Jacob Sorber. **The Amulet Wearable Platform: Demo Abstract**. In *Proceedings of the ACM Conference on Embedded Network Sensor Systems (SenSys)*, pages 290–291, November 2016.

Travis Peters, Srikanth Varadarajan, and Reshma Lal. **Poster: Security in IoT: What is IoT Security, Really?!** *Intel Labs Open House*, September 2016.

Travis Peters, Srikanth Varadarajan, Pradeep Pappachan, and Reshma Lal. **Poster & Demo: Protecting Bluetooth Input from Malware**. *Intel Labs Open House*, September 2015.

Travis Peters, Srikanth Varadarajan, Pradeep Pappachan, and Reshma Lal. **Poster: Trusted I/O and Bluetooth Devices**. *Intel Labs Intern Poster Show*, August 2015.

Travis Peters. **An Assessment of Single-Channel EMG Sensing for Gestural Input**. Technical Report TR2015-767, Dartmouth Computer Science, September 2014.

Talks & Presentations

[**Workshop Talk**] BASTION-SGX: Bluetooth and Architectural Support for Trusted I/O on SGX. Workshop on Hardware and Architectural Support for Security and Privacy (HASP) at the International Symposium on Computer Architecture (ISCA), Los Angeles, California, June 2018.

[**Workshop Talk**] Physical Emanations and Potential Applications. Annual Trustworthy Health and Wellness Workshop, University of Illinois at Urbana-Champaign, Champaign, IL, September 2017.

[**Invited Talk**] An IoT Survey: Security, Privacy, and Safety in the Future of IoT. Intern Tech Talk Series, Intel Labs, Hillsboro, Oregon, September 2016.

[**NSF Research Outreach**] Fitbit Project: Discussing the Fitbit System, Data, and Security & Privacy Awareness. Hanover High School (Statistics Class), Hanover, New Hampshire, May 2015.

[**Invited Talk**] Delivering Secure Bluetooth Device Input to a Trusted Execution Environment. Intern Tech Talk Series, Intel Labs, Hillsboro, Oregon, September 2015.

[**Poster Presentation**] Security in IoT: What is IoT Security, Really?! Intel Labs Open House, Intel Labs, Hillsboro, Oregon, September 2016.

[**Poster Presentation & Demo**] Protecting Bluetooth Input from Malware. Intel Labs Open House, Hillsboro, Oregon, September 2015.

[**Poster Presentation**] Trusted I/O and Bluetooth Devices Intern Poster Show, Intel Labs, Hillsboro, Oregon, August 2015.

Technical Skills

Programming Languages: Python, C, Java, Javascript, Matlab, x86 assembly, Bash, Ruby, SQL, LaTeX, HTML/CSS. ***Software Development & Prototyping:*** OSX, Linux, Android, iOS; Linux and Android Bluetooth stacks; Raspberry Pi, Arduinos, and other custom platforms (e.g., Amulet); Git, SVN, Perforce; Vagrant, Docker, Chef. ***System & Software Inspection & Diagnostics:*** software inspection, e.g., GDB, dtrace, strace, ptrace, perf; physical inspection, e.g., oscilloscopes, spectrum analyzers. ***Data Collection & Analysis:*** Wireshark, GNU Radio, Jupyter, MATLAB. ***Wireless and Software Defined Radios (SDRs):*** Ubertooth; USRP, LimeSDR; GNU Radio. ***Databases & Web Frameworks:*** MySQL, MongoDB; Node.js.

Awards & Honors

Funding Acknowledgements *2014 - 2018*

My research as a PhD student has been conducted under the guidance of my advisor, Dr. David Kotz, and has primarily been funded by the following awards:

NSF SaTC, $10,000,000 (Dartmouth share $4,000,000) for 2013-2018.
Enabling trustworthy cybersystems for health and wellness (THaW). PIs: David Kotz (Dartmouth), Kevin Fu (University of Michigan), Carl Gunter (University of Illinois at Urbana-Champaign), and Avi Ruben (Johns Hopkins University) Learn more at `thaw.org`.

NSF CNS, $2,712,286 (Dartmouth share $1,815,866) for 2013-2018.
Computational Jewelry for Mobile Health (Amulet). PIs: David Kotz, Ryan Halter, Andrés Molina-Markham, Sarah Lord (Dartmouth); Jacob Sorber and Kelly Caine (Clemson). Learn more at `amulet-project.org`.

Best Teaching Assistant Award[1], Department of Computer Science, Dartmouth College *2014 - 2015*

Outstanding Graduate Student Teacher, Dartmouth Center for the Advancement of Learning *April 2015*

Graduate Student Teaching Award[2], Dartmouth College *2013 - 2014*

Dartmouth Fellowship, Dartmouth College *2013 - 2014*

Oscar Edwin Olson Scholarship, Western Washington University *2012*

Outstanding Winner, Frank Giordano Award[3], Contest in Mathematical Modeling *2012*

Kaiser Borsari Scholarship, Western Washington University *2011 - 2012*

Giusti Scholarship, Western Washington University *2011 - 2012*

Leadership & Volunteer Experience

Topo Ambassador, Topo Athletic *2018 - present*

Co-Webmaster, Upper Valley Running Club *2018 - present*

Lead Sunday School Teacher, Christ Redeemer Church *2014 - present*

Assistant Track Coach, Hanover High School *Winter-Spring 2017*

Free Geek Build Volunteer, Free Geek (Portland, OR) *Summer 2016*

Assistant Track Coach, Hanover High School *Spring 2016*

Organizer & Facilitator, Graduate Student TA Workshop *December 2015*

Graduate Student Council Rep., Dartmouth College Computer Science *2013 - 2015*

Graduate Student Web Team, Dartmouth College Computer Science *2014 - 2015*

Lead Teacher & RK Coordinator, Redeemer Kids at Redeemer Church *November 2011 - June 2013*

Vice President of Business & Operations, Associated Students of WWU *June 2011 - June 2012*

Chair, AS Management Council & AS Facilities & Services Council *June 2011 - June 2012*

Vice-Chair, AS Board of Directors & AS Budget Committee *June 2011 - June 2012*

Member, Academic Honesty Board *June 2011 - June 2012*

Big Brother, Big Brothers Big Sisters of Whatcom County *September 2010 - September 2011*

4-year member of Varsity Track & Field, Western Washington University *2008 - 2012*

4-year member of Junior Varsity Cross Country, Western Washington University *2008 - 2011*

References

Available upon request.

[1] An award voted on by all CS faculty at Dartmouth.
[2] An award given to only three graduate students across Dartmouth.
[3] Less than 3% of teams (10 out of more than 3,600) are selected as Outstanding Winners of the contest.

Teaching Statement

Travis W. Peters
Department of Computer Science
Dartmouth College

I took my first computer science class shortly before I turned 20. I wasn't like many of my peers: I wasn't raised programming Apple II's; I didn't grow up obsessed with Bill Gates or Steve Jobs. And after struggling through my first few computer science classes, I couldn't escape the thought that I didn't belong. But something kept me coming back to computer science classes. I have since realized that what kept me coming back is that when I found computer science as an undergraduate, I didn't just find a new subject to study or a potential skill for a future job – I found a vehicle by which I could explore virtually any interest I'd ever had. This vehicle is precisely what motivates my passion for teaching and lies at the foundation of what I try to share with students in my role as an educator.

Teaching Experience

Prior to coming to Dartmouth I served as an educator in various settings and capacities. For example, I spent a summer as a teaching assistant at a small private school with children who required more individualized attention and education plans. I have also worked as a tutor, teaching assistant (TA), and instructor in college mathematics and computer science courses. For the purpose of highlighting my teaching experience, I'll concentrate on more recent examples from my time at Dartmouth.

In my first year I had the opportunity to be a TA for CS1 (Introduction to Programming and Computing) and CS10 (Problem Solving via Object Oriented Programming). I held regular office hours, and organized a staff of undergraduate TAs who held weekly recitation section – small learning groups with instruction and short activities to reinforce concepts covered in class. A major part of my role as a TA in these classes was to mentor undergraduate TAs: helping them lead small group discussions, and giving them feedback on how they graded assignments. Throughout the year, I worked with *many* incredible people – students, fellow TAs, and professors. Together we took more than 500 students on a whirlwind tour of foundational and popular ideas in computer science including data structures, algorithms and algorithmic analysis, program design, machine learning, networking, graphics, mobile device programming, and security and privacy.

After receiving exceptional feedback in my first year as a TA, I received multiple awards. Among them are: the *Outstanding Graduate Student Teacher Award* – an award in which undergraduate students nominate graduate student TAs; the *Best Teaching Assistant Award* – an award in which professors in the Computer Science Department vote and select only one CS graduate student; and the *Graduate Student Teaching Award* – a college-award in which professors nominate students across the college. (In the year that I won this award, only three students across the entire college were selected to receive this award). In addition to these awards, I was also offered my favorite award of all of these: the opportunity to be a course instructor at Dartmouth. For this class of more than 100 students I prepared and delivered interactive lectures, held office hours, met 1-on-1 with students, coordinated with undergraduate deans to make accommodations for students with special learning needs, managed a staff of 1 graduate and 12 undergraduate TAs, organized recitation section topics and activities to reinforce learning outside of class, and wrote exams and assignments to asses my students. *What an incredible learning opportunity!* Without a doubt, this experience was remarkably defining. In fact, in the next section I discuss my teaching philosophy, which has absolutely been shaped by this experience. Throughout this text I draw on student comments[4] that highlight my philosophies put into action in my teaching.

Since teaching CS10, I have had to spend more time focused on research, but I have tried to stay active in developing my teaching abilities and pursing more teaching opportunities. An example worthy of note is my time as a TA for CS50 (Software Design and Implementation) in Spring 2016, which was taught by my PhD advisor, David Kotz. As a TA I delivered guest lectures, composed class notes, and designed exercises for students. I also worked closely with my advisor to develop a large culminating team project for the course. For this project, we purchased 50 Pebble smartwatches,

[4]The complete collection of reviews from my term as an instructor of CS10 can be made available upon request.

and had teams of students design and implement software to carry out a cooperative and competitive game.[5] The software spanned multiple platforms: some ran on the pebble smartwatches, some ran on laptops, and some ran on servers; to support the project I designed and built an infrastructure that allowed their pebble smartwatches to communicate with their servers via a smartphone app. Through this project, my advisor and I were able to teach students about wearable devices and event-oriented programming (which closely resembled one of our research projects known as *Amulet*[6]). This project also promoted good software engineering practice; we organized design reviews and implementation reviews to provide early feedback to students on their plans and code. At the end of the class, each team delivered a live demonstration of their project, as well as a testing plan that detailed how students thought about testing the various components of their project.

Teaching Philosophy

My teaching philosophy stems from my past experiences, both as a student and a teacher. Informally, there are simple and fairly commonsensical philosophies that guide my teaching. Among them are *talk with students as you would like to be talked with – positive engagement comes through mutual respect,* and *care about the interests of students – new concepts make more sense when related to something familiar,* and *show off your own passion and enthusiasm – if you are bored it is likely they will be as well.* Over the years, these philosophies have been refined and molded into five pillars that make up my teaching philosophy and provide a framework for my pedagogy.

Growth Mindset. First and foremost, I believe in having a *growth mindset*. Specifically, I believe in the science that shows that people who are taught that they can learn and improve, see drastic impacts on their motivation and achievement.

I have a growth mindset in my teaching. I know that I am not yet the best teacher that I can be; however, I also know that the teacher that I am today is not the teacher I will be months and years from now. To facilitate my own growth, I seek input from peers and more seasoned educators. For example, In my years at Dartmouth, I have attended countless workshops at the Dartmouth Center for the Advancement of Learning (DCAL)[7] – a hub for scholar-educators to share resources and learn about teaching practices. After attending many of DCAL's workshops I also worked with more experienced TAs to organize a "TA Orientation" workshop.[8] The workshop we designed taught new graduate student TAs from a range of departments about resources on campus, time management strategies, how to create inclusive learning environments, and how to navigate "tricky situations" (such as owning your mistakes, friendships with students, and disputes between group members); we even facilitated micro-teaching opportunities where TAs could practice teaching a short lesson relevant to their field, and also learn strategies for effectively giving and receiving feedback. Throughout this workshop, we used case studies and small group discussions intermixed with mini-lessons prepared by the organizers to actively engage workshop participants.

In addition to feedback from peers and seasoned educators, I actively seek feedback from my students. For example, as an instructor of CS10 I collected midterm feedback through anonymous surveys. These surveys helped me identify what was and wasn't working in my class, and enabled me to make informed improvements based on student feedback. (Collecting midterm feedback enabled me to make improvements along the way, rather than not receiving structured feedback until after the course ended.) One change that I made as a result of the survey results was to adjust the days and times that TAs were available and increased the number of office hours I offered on certain days. I also tweaked the style of lectures to use more pre-prepared slides and code snippets; because our class was in a large auditorium and my hand writing is small, students found slides more helpful than using whiteboards in class. I also shared my class notes and slides, which gave students more access to out-of-class study resources.

My strong belief in having a growth mindset also translates to *how* I teach my students. In my teaching, I strive to engage students and interact with them in such a way that builds trust and fosters learning; this is a nice segue to my remaining pillars.

[5] Project specs. are online at `https://traviswp.github.io/classes/pebble-project-intro/specs-2016/`. Notes for a related guest lecture can be found online at `https://traviswp.github.io/classes/pebble-project-intro/`.

[6] Information about the Amulet project is online at: `https://amulet-project.org/`

[7] Visit DCAL's website at: `https://dcal.dartmouth.edu/`

[8] Resources from the TA Orientation workshop can be made available upon request.

Engagement and Active Learning. I love many facets of computer science, but a challenging reality exists: it can be dry and outright boring at times! One cannot avoid discussing concrete matters, such as syntax and low-level details, altogether; in my teaching, however, I aim to weave this information into compelling stories, helpful analogies, and culturally and temporally relevant lectures and projects. To this effect, one student commented:

> *"In my eyes, someone providing help for this course should balance explanation of concepts with demonstration of code/syntax/technique. Travis was very helpful in both areas, he did an excellent job."*

Furthermore, I believe that teaching needs to be active; we learn best through interaction, experience, exploration, and even failure. In my classes, I regularly ask questions to engage students and assess whether students are following along. I also try to incorporate activities where students can be active. For example, as a TA for CS50 (a programming-intensive software engineering course), I designed a lecture and activities on debugging programming logic and memory leaks with GDB and Valgrind.[9] I taught students basic concepts and commands, then let them explore a buggy program. Using their new tools, students were able to discover that the buggy program contained a buffer overflow. This activity also created an opportunity to share my own research interests in security and illustrate to students how programming errors can be exploited in practice. This class was so successful that I've been invited back multiple times to give this guest lecture in other classes.

Equity and Inclusivity. I strive to create environments that are welcoming to people of all backgrounds and skill-levels, and one that encourages students to ask questions – in my classrooms and in my office, *there is no such thing as a stupid question.* My own past struggle with feeling like I did not belong in computer science has helped me to relate and connect with many students, especially introductory students. One student stated:

> *"...He never made me feel stupid for asking questions that the other kids in class, who have coded in 1000 different languages already, understood just by attending lecture."*

Differentiated Instruction. I don't believe it is enough to be competent in a particular area to teach well. Rather, I believe instruction should incorporate knowledge of students' background and learning styles. Indeed, some of my most fond memories from my time as a student have been interactions with professors that did just this; interactions where it was evident that they paused to take time to digest my question, incorporate what they knew about me, and *only then* respond. Because of this, I always try to take time to really listen to a student, consider who they are and what I know about them, identify where the problem is in their understanding, and then craft a response. One student stated:

> *"I found Travis to have a clearer idea of what the confused CS student was asking about and provided specific guidance."*

Trust. I attempt to establish trust with students by conveying my own work ethic, my own fallibility, and the fact that I am committed to them as people and their work as students. In my experience, this can be as simple as demonstrating that you care about their success and that you are willing to do what you can to see that they can be successful. One student commented:

> *"Now, Travis was amazing. He had this motto, 'I won't leave you with broken code.' He stuck by this, always..."*

At Dartmouth I have applied these philosophies with the specific goals of gaining the trust of my students, instilling a problem-solving mindset in them, and helping them to establish a strong foundation in computer science concepts and techniques. While doing this, I have continually invited feedback from students and professors; constantly looking for ways to improve how I teach and foster learning. And so far, I am encouraged by the highly positive feedback that I have received from students and professors.

[9] My notes and the activities are available at `https://traviswp.github.io/classes/debugging-gdb-valgrind/`

Teaching Interests

In the future, I look forward to teaching both introductory and advanced courses. I am comfortable teaching a variety of levels and subjects that intersect with my area. These include data structures, algorithms, operating systems, computer architecture, networking, and security and privacy. I am especially passionate about teaching introductory courses as I yearn to share with others – especially those new to computer science – about the exciting opportunities within the field, as well as to help those that will not stay in computer science, take something new with them back to their primary disciplines. I am also interested in developing courses that more closely relate to my research in computer security and mobile health. These classes could span topics in which *"the software meets the silicon,"* such as mobile and wearable systems, pervasive computing, trustworthy computing, and security and privacy in the Internet of Things.

Teaching Non-Computer Scientists

I want to emphasize my interest in teaching students that are not (yet) computer scientists. Earlier in this statement I shared that my interest in computer science was born out of an understanding that it is a vehicle for learning. Indeed, to me, computer science is so much more than circuits and logic, and algorithms and data structures. Computer science enables people to solve problems; more important, it enables people to solve the problems that are important to them.

This definitely applied to me. When I began graduate school I had no formal education in healthcare yet problems in this space were important to me because my mother works in healthcare and I was interested in the work she did to help people. Furthermore, I had no formal education in computer security, but my interest in mathematics, coupled with growing societal concerns around increased dependence on technology, piqued my interest. These are the problems and domains that have been important to me; and I've been fortunate to know and teach people who likewise saw computer science as a vehicle for learning and solving problems that are important to them. For example, I've helped my father-in-law – a grade-school teacher – compose a website dedicated to helping his students learn how to be better writers. I've helped students work on programs that allowed them to study constellations using data collected at Dartmouth's observatory. And I've helped behavioral scientists design smartphone apps that work with wearable sensors, which enabled them to study smoking addiction.

In the future, I hope to be involved in more opportunities where I can share computer science with non-computer scientists, and teach them ideas, techniques, and technologies from computer science that help them solve all sorts of exciting problems.

Future Goals & Closing Thoughts

I feel fortunate to have received positive feedback on my teaching thus far, but I know that the job of an educator is never done. Looking forward, my goals are mostly oriented around further developing my abilities as a teacher, establishing connections between my research and classes I'll teach in the future, and learning more about creating environments that are inviting to diverse groups of people.

While at Dartmouth, I have continued to attend class lectures and invest in professional relationships with professors. These relationships provide opportunities to study the ways different professors present their ideas, connect with students, as well as manage their classes and course staff. I have also persisted in my efforts to maintain relationships with former students, as well as forge new ones. These serve as opportunities to practice mentoring, teaching, and ultimately establishing meaningful connections, with people across the various dimensions of culture, socio-economics, race, ethnicity, age, religion, gender, sexual orientation, physical and mental abilities, or personal views.