Teaching System Design in Experiential Learning: Building a Fitness Wearable at Home

Dr. Ramsin Khoshabeh, University of California, San Diego

Ramsin Khoshabeh received his PhD from the ECE department at UC San Diego in 2012, specializing in computer vision and machine learning for medical applications. He currently serves as the Director of the ECE Makerspace at UC San Diego. In addition to overseeing the operations of the lab, Ramsin also teaches numerous experientially-focused courses covering topics such as basic electronics and prototyping, wearable sensors, Python programming, full-stack web development, real-time signal processing, machine learning and vision, human-centered product engineering, and even agile business planning. Prior to entering the workforce at UC San Diego, Ramsin is part of several startups and consults with a number of local companies on computer vision, machine learning, and blockchain technologies.

Vikash Gilja

Teaching System Design in Experiential Learning: Building a Fitness Wearable at Home

Abstract

At our university, the ECE department has striven over the last few years to provide undergraduate students with an educational experience that far exceeds the expectations of hiring managers. We surveyed students and employers to understand where the gap exists between new graduates and highly qualified engineers. New graduates frequently struggled at attaining the best internships. Even before they graduated, many of them started to seek out opportunities but often in vain. Furthermore, most new hires had to go through a season of training before they could become contributing employees. As a result, we wanted to design a course that would help us address the research question: "How can we deliver an engineering education that provides students the skills they need to succeed in the workforce?"

By genuinely listening, we discovered a number of key insights which led to a highly successful course where students rapidly design hardware and software to interface with the world. In this paper, we discuss our motivations, the design of the course, what we have learned from teaching the course, and where we see the future of experiential education heading, especially in light of the COVID-19 pandemic and the need for highly effective remote instruction. We believe that the model we have created in this classroom experience successfully prepares students for the rigors of an engineering career.

Our ECE department has a rich history of exemplary theoretical teaching, with a strong emphasis on research, but undergraduate students felt a void in how to apply that knowledge into engineering practice, especially in future careers. This is why in recent years we have strongly focused on experiential learning in all four years of the undergraduate experience. We developed courses for entering freshmen and capstones for graduating seniors, but we did not have anything for students in the heart of their university experience, particularly for those uncertain of their future aspirations. This was a driving force behind the formation of this course. The goal of the class was, and still is, to offer undergraduates experience with real-world data, teach them to work with a complete system, and provide them a contextual basis in which to apply their theoretical knowledge. This goal was established after careful consultation with our corporate affiliates and alumni. As a result, the course today has students build a fitness wearable from first principles. During the journey, they attain foundational Python software development skills and are exposed to many facets of ECE curriculum. In their final project, they repurpose their wearable to address a new, unrelated problem so as to be challenged to be critical thinkers working on open-ended problems – a highly sought-after skill by employers we surveyed.

Due to the modular, often self-paced nature of the course, it has had a serendipitous outcome during the pandemic – namely, while being a highly hands-on course, it actually works extremely

well in settings of remote instruction. Feedback from students has been surprisingly positive as they have had to work on their project kits from their homes rather than in the lab setting. Since much of their instruction in their other classes has shifted to lectures offered via video conferencing software, any opportunity to actually work with their hands has led to marked excitement and eagerness to participate in class, as has been directly observed by us.

The focus of this paper will be to breakdown the course curriculum, demonstrate how it offers students a unique learning experience, and illustrate the effectiveness of the material even during remote instruction.

Introduction

As the abstract suggests, designing hardware and software to interface with the world does not mean that this is just another embedded systems class. When fellow engineers and academics initially hear about this course, their immediate reaction is often to run to this kind of thinking. This is only natural as their own personal experience was reflective of that style of education more often than not. Unfortunately, for those of us who have had to build consumer products, we very well know that a microcontroller and some sensors does not a system (nor a product) make!

Teaching students about embedded systems is valuable knowledge, but the objective of this class goes beyond that oversimplification. When designing this course for our students, we observed a three-fold deficiency in the student body. Firstly, ECE students lacked hands-on experience building a complete system. We had an excellent theoretical program teaching students the fundamentals of ECE, but all of our lab-oriented classes focused on small, self-contained assignments, not translating well to generally-usable skills. We desired students to have an experience where they worked with real-world data (e.g., biosignals) to design and build a complete system from the ground up. Secondly, ECE is an extremely broad discipline. Asking students to decide their future career path without giving them at least a taste of what sorts of things they could be doing is a daunting task. This leads to a disillusioned undergraduate experience for many students as they fumble through their journey of self-discovery. As a result, our second desire was to give students an experiential sampling of many of the ECE depth focuses early in their academic careers, so that they would have a better sense of where their passions lie. Finally, based on feedback from our industry partners and alumni, we saw that the students performed very poorly in software design. When they were tasked with writing a small script to accomplish a specific goal (e.g., computing the Fibonacci sequence), students performed just fine. However, when given a larger design specification and asked to build a complete end-to-end system integrating both hardware and software, students did not even know where to begin. Some might argue that those skills should belong only to computer scientists, but that is simply a fallacy. For the vast majority of engineering professions today, good programming skills are no longer an option but a prerequisite.

With these insights in mind, we designed a sophomore-level course that would have students build a complete system from start to finish, expose them to a broad spectrum of the ECE areas of focus, and require that they apply architectural thinking in designing and applying good software development principles. Furthermore, we designed the course to be a mostly-flipped classroom to maximize student engagement and support. Considering that it is a hands-on course, we went to great lengths to make sure that the student project kits would be easily accessible and not require the students to utilize a laboratory environment.

Related Works

When we consider hands-on education using embedded systems, we stand on the shoulders of giants. In designing this course, we worked closely with our Teaching and Learning Commons to incorporate the latest pedagogical research as well as looked for innovative approaches and technologies being utilized by other institutions. For example, [1] presents a novel approach to offering embedded systems labs remotely by incorporating a cloud-based camera system with which students can interact. When considering offering our course remotely, we contemplated such an approach, but we concluded that it would be most beneficial to the students to send them individual kits. We understand that not all courses will work well in that format, but we found it to be the best option for our students.

In our planning stages, we explored the approaches of other prominent universities. Notably, UC Berkeley highlighted some of the findings that we consider to be valuable in [2]. In designing their courses on embedded systems, they stressed critical thinking about the system design rather than the embedded system specifications. They also structured the material so that a sequence of exercises would culminate in a design-focused capstone project. We modeled this approach very closely in designing our lab and final project structure.

In [3], El-Abd conducts a survey of courses utilizing Arduino-based embedded systems. The survey highlights that excellent assessment methods for such experiential courses tend to be some combination of presentations with associated demonstrations. Our course follows this modality, where student grades are largely based upon their project presentations and demonstration of their functioning systems. The paper also finds that hands-on courses using Arduino-like embedded systems lead to increased motivation and interest on the part of the students. Lastly, the paper highlights that the embedded system should serve industry needs in order to properly prepare students for the workplace. In our course, we do use the Arduino IDE for programming the microcontroller, but we actually chose to work with an ESP32 rather than a typical Arduino board because of this very exact reason. We wanted students to believe that they were actually preparing for the workforce.

Similarly, [4] explores the offering of a hands-on circuits course at a large scale using the Coursera platform. The authors conclude that even on an online platform, having labs as part of the curriculum enhances overall student performance. Additionally, the labs improve student confidence in the topics of the course. While we have not yet scaled the course to the level of being offered on an online platform, these findings really resonated with us as we contemplated how to structure the course to marry theoretical and practical knowledge together into a course.

These findings are echoed in other works, such as [5] that highlights the importance of experiential ECE education in making well-rounded engineers or [6] where the authors present that combining abstract theory with practical application leads to enhanced pedagogical learning.

A well-planned course design is important for a successful student experience, both in and out of pandemics, as is echoed by [7]. Thus, we have made every effort to follow solid instructional design models [8] in putting together our curriculum, and we believe that the course structure and content we have assembled truly delivers a unique educational experience.

The Course Curriculum

The curriculum has gone through numerous iterations as we have continuously solicited feedback and modified the course content to best align with the desired learning outcomes of this class. While initially still focused on practical experiential learning, the course has gone through a number of revisions as we have learned better techniques for equipping the students.

The current structure of the material is the result of working closely with our Teaching and Learning Commons, highlighting the learning outcomes we discussed above, and then actively trying to support the students through engaged learning.

Labs Structure

There exist a total of seven weekly lab modules. Each lab introduces a set of related topics that build upon the material from previous labs. The labs used to be two weeks in length, allowing for flexibility in fitting with the students' personal schedules, but we discovered that the flexibility actually motivated procrastination. Instead, to encourage a strong work ethic, we shifted to a weekly schedule. The labs are gamified so that the next lab will not unlock unless a student completes the prior lab first. This incentivizes them to get something working, even if their solution is not ideal.

Over the course of the seven labs, the students end up building a fully functional (although not nicely packaged) fitness wearable – it tells the time, retrieves local weather information to display on the screen, measures step count and heart rate, and offloads all heavy computation wirelessly over Bluetooth onto their computer.

Figure 1 depicts the overall system that the students build during the weekly labs, while the following bulleted list covers at a high-level what each lab emphasizes. At first glance, this seems like a daunting amount of material to cover in just 7 weeks. However, the way that we have structured the course, the students are only presented with as much of the theoretical material as is needed to accomplish the task at hand. The goal is to expose them to a wide breadth of topics so that they can discover their own affinities for future study and exploration, while also being equipped to complete the system design.

Weekly Lab Breakdown

- Lab 1 – Microcontroller Basics, Digital & Serial Communication, Git Version Control
- Lab 2 – I^2C Communication, OLED Visualization, Analog I/O, Sampling
- Lab 3 – Python Basics, Bluetooth Communication
- Lab 4 – Object-Oriented Programming, Scientific Visualization
- Lab 5 – DSP with Python, Building a Pedometer
- Lab 6 – Optical Pulse Sensing, Building a Heart Rate Monitor
- Lab 7 – Machine Learning for a Robust Heart Rate Monitor

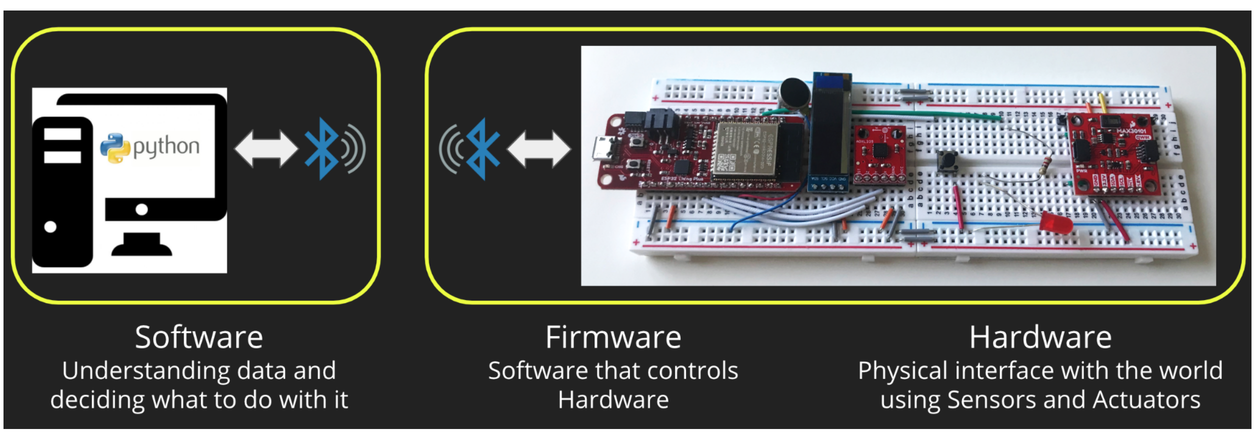

Figure 1. Lab Overview: Students assemble the hardware depicted on the right and build the firmware and software that produces a functional fitness wearable wirelessly measuring heart rate and step count and providing users feedback with both a visual display and haptic vibration.

Final Project

Upon completion of the labs, the students must complete a grand challenge final project. The assessment criterion is to see if students can generalize well the knowledge that they have attained, both in the hardware utilization and in the software implementation. In previous years, we had left the project open-ended for them to explore various design challenges. However, due to the COVID-19 lockdowns, we standardized the project so that all students work on the same design challenge.

Each team of two students is given a modified video game (Space Invaders) that can receive controller commands via software input. Their task is to design and build a wireless controller for the game given the knowledge they have attained in the labs. They are free to use all hardware, firmware, and software attained through the labs, but their solutions must be functional, unique, and also well-designed.

As a final assessment, the students submit a video demonstrating their operational controller where they also discuss their implementation details. Then they push all of their well-documented codebase to version control, just as they would be doing if they were working as professional engineers.

A Mostly-Flipped Classroom

We offer two 3-hour blocks of time for students each week as lab and lecture time. The content of the course is offered in a flipped manner, so the bulk of that class time is devoted to supporting the students on their projects. For approximately the first hour, we will review the project progress, do activities, or discuss common difficulties experienced in the build and design stages. For the remainder of the class time, we allow the students to work in a self-paced manner, repeatedly emphasizing the importance of timely work ethic.

Each week the students are given access to that week's lab prior to the meeting times. Every lab contains four sections discussed below:

1. Video lectures
    a. These are brief recordings covering a specific topic. Oftentimes, each lab will have 2-3 associated lectures. We have found that students respond well to material when it is presented in smaller, focused segments. They often refer back to these lectures, even after completing the course.
2. Quiz
    a. The quizzes are simple and serve to ensure that students actually watch the recorded lectures. In order to unlock the tutorials and challenges for the lab, they

must pass the quizzes. The goal is not to use them as a knowledge check but to deter students from simply trying to skip ahead and impair their progress.

3. Tutorials
    a. The tutorials typically go hand-in-hand with the video lectures. They demonstrate by example specific techniques that the students will need to learn to complete the challenges. For instance, when a student needs to design a pedometer, we first go through a tutorial covering how to interface with the analog sensor to read the raw data. The tutorials guide them through the process that they then need to utilize to accomplish the task.
4. Challenges
    a. The challenges are the most critically assessed portions of the labs. Once students complete the previous three sections, we make sure that they have all the prerequisite knowledge and tools to accomplish the challenge.
    b. It is in the challenges where we assess critical thinking, creativity, and the effective application of knowledge gained from the lectures and tutorials.

Results

While we still need to conduct a thorough quantitative assessment on the efficacy of this course, based on the feedback we have received from students thus far, the format of the course is working incredibly well.

We have not yet conducted a formal survey to gauge the success of students after they complete the course and go on to their careers. However, we do conduct anonymous Course And Professor Evaluations (CAPE) assessments after every term that the class is taught. We have received nearly unanimous support that the students found the material relevant, engaging, and important for their futures. The most critical feedback we have received was in the early offerings of the class where students felt they could not keep up with the workload, but we quickly adjusted to address those concerns.

We also complete an exit poll of all students after their final project has been submitted. When asked if they felt that the course met its mission (of giving them confidence as an engineer, exposing them to a variety of ECE subject matter, giving them the opportunity to think critically and creatively, and making engineering fun), 100% of the respondents believed that it did. When asked what they thought was the best part of the class, we received a diverse set of answers ranging from: the ability to prototype, building a system, working with real-world issues, diving deep into programming with Python, etc. This reveals to us that the course is working well in helping students discover what sub-disciplines of ECE really strike a chord with them as they seek to pursue their careers.

In addition to the anonymous reviews and exit surveys, we have also kept up personally with many of the former students who have gone on to be successful engineers and researchers. Interestingly, while preparing this draft for submission, we received an email from a student who took the class last year during the initial wave of the COVID pandemic. While many classes struggled to maintain student engagement, we found that students actually loved the ability to keep themselves busy with a project while having to be isolated at home. We omit the name of the student, but the body of the message reads as follows:

"I am working on creating a small data acquisition device that can read resistance values of a wearable sensor fabric (for a research project)! I am using a lot of the techniques we learned... I wanted to say thank you for teaching that class because it was incredibly valuable and it's the first time I feel like I can use the knowledge learned in a course for a real-world application!"

We are planning on conducting a longer-term assessment of students in the near future, but we believe that the responses we have received thus far are indicative that our curriculum is working well in producing the intended learning outcomes.

Conclusion

While we are teaching our students, we have learned a great deal ourselves. We have learned that most students, particularly in the field of ECE, are unclear about their career paths and need experiential exposure to the possibilities. We have also learned that just because students can get A's in classes does not mean that they have mastery of the material. They really need to apply it in order to internalize the knowledge. Once they internalize the knowledge, they continue to build upon what they learn, particularly in their own personal pursuits as evidenced above.

Flipping the classroom material was a serendipitous undertaking for us during the pandemic, but it revealed that such a structure provides much more personal time with students as they work to design, build, and debug their systems. Establishing personal relationships in turns leads to greater engagement by the students. Additionally, even though the class requires hands-on project work, offering the instructional material in a flipped classroom setting alleviates some of the scaling issues commonly encountered in project-based courses.

While we have had to live through some devastating times during the pandemic, not everything has been bleak. Remote projects invigorate and embolden Zoom-fatigued students, giving them an opportunity to actually build something meaningful and feel connected to their classmates. In the same vein, “home labs” do not have to go away when COVID finally does. Students’ productivity increases when they can work both at home and in the campus lab. None of us anticipated a worldwide pandemic to ever take place in our lifetimes, but teaching this course during the pandemic has shown us that the future of experiential learning in a Post-COVID era is bright indeed.

References

1. I. Angulo, L. Rodrìguez-Gil, and J. Garcìa-Zubìa, “Scaling up the Lab: An Adaptable and Scalable Architecture for Embedded Systems Remote Labs,” in IEEE Access, vol. 6, pp. 16887-16900, 2018. doi: https://doi.org/10.1109/ACCESS.2018.2812925.
2. E. A. Lee, S. A. Seshia, and J. C. Jensen, “Teaching embedded systems the Berkeley way,” In Proceedings of the Workshop on Embedded and Cyber-Physical Systems Education (WESE '12), Association for Computing Machinery, New York, NY, USA, Article 1, 1–8, 2012. doi: https://doi.org/10.1145/2530544.2530545.
3. M. El-Abd. A Review of Embedded Systems Education in the Arduino Age: Lessons Learned and Future Directions. International Journal of Engineering Pedagogy, 7(2), 79-93.
4. B. H. Ferri, A. A. Ferri, D. M. Majerich, and A. G. Madden, “Effects of In-class Hands-On Laboratories in a Large Enrollment, Multiple Section Blended Linear Circuits Course,” Advances in Engineering Education 5 (3), n3, 2016.
5. S. C. Hu, “A wholesome ECE education,” in IEEE Transactions on Education, vol. 46, no. 4, pp. 444-451, Nov. 2003, doi: https://doi.org/10.1109/TE.2003.818753.
6. E. Taslidere, F. S. Cohen and F. K. Reisman, “Wireless Sensor Networks—A Hands-On Modular Experiments Platform for Enhanced Pedagogical Learning,” in IEEE Transactions on Education, vol. 54, no. 1, pp. 24-33, Feb. 2011, doi: https://doi.org/10.1109/TE.2010.2041235.
7. C. Hodges, S. Moore, B. Lockee, T. Trust, and A. Bond, “The Difference Between Emergency Remote Teaching and Online Learning,” Educause Review, 2020.
8. R. M. Branch and T. A. Dousay, “Survey of Instructional Design Models,” Association for Educational Communications and Technology, (AECT), 2015.