The Integration of Ethical Design into Introductory Software Development Coursework

Technical Report
Presented to the Faculty of the
School of Engineering and Applied Science
University of Virginia

By

Justin Ngo

February 28, 2022

On my honor as a University student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments.

Signed: _____Justin Ngo_____

Approved: _____ Date _____
Daniel Graham, Department of Computer Science

The Integration of Ethical Design into Introductory Software Development Coursework

CS4991 Capstone, 2021

Justin Ngo
Computer Science and Engineering Business
University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
jmn4fms@virginia.edu

ABSTRACT

Ethical thinking in Computer Science coursework is critical because many students in Computer Science programs become software engineers— responsible for shaping society through their products and making informed judgements in their work based on legal and ethical principles. However, the University of Virginia's Computer Science program lacks a strong cultivation of ethics. I examine the ethical computing concepts discussed in the Spring 2021 iteration of CS 3501 Everyday Ethics and Quotidian Quandaries for Computer Scientists, and the basics of software development taught in the Fall 2019 iteration of CS 2110 Software Development Methods, to propose improvements to the teaching of ethics at UVA. I suggest that the teaching of ethics and software development should be integrated and taught early in the curriculum as opposed to the current and limited approach at the end. The process of integrating ethics into the Computer Science curriculum reinforces the idea in students that ethics and engineering go hand in hand. It also creates a more welcoming environment for students of different backgrounds to promote diversity in a field that is actively seeking it.

1 Introduction

In recent years, there has been an increasing number of news articles about software developers designing algorithms that can easily stray into biased or discriminatory behavior. Algorithms deciding employment, policing neighborhoods, assigning risk scores, skewing public perception of news, and determining which enemy combatants should be killed are being brought to the forefront of public attention. [1] Therefore, Computer Scientists and Computer Science students no longer have the luxury of ignoring the consequences of their work, and it is now more important than ever that students are properly equipped to build ethical systems.

Current UVA Computer Science coursework approaches teaching the ideas of ethics and introductory software development in isolation. Students in the School of Engineering and Applied Sciences are exposed to ethics in a limited manner through three required Science, Technology, and Society courses: STS 1500, STS 4500, and STS 4600. STS 4500 and STS 4600 are only taught to students in their final year of study. Computer Science students in both the College of Arts and Science and School of Engineering and Applied Sciences are exposed to ethical concepts in either standalone ethics classes such as CS 3501 Everyday Ethics and Quotidian Quandaries for Computer Scientists, or later courses that contain limited integration of ethics such as in CS 3240 Advanced Software Development. [2] These courses are optional and not required for Computer Science students in the College of Arts and Sciences. These courses are also not intended to be taught in the first two years of a student's Computer Science coursework.

This isolated approach to teaching ethics is flawed because it does not adequately prepare students for the impactful engineering projects they will encounter in the real world. Real world projects require ethical consideration while being worked on and cannot be considered only as an afterthought. In addition, although the teaching of technical concepts and ethics separately carries the advantage of greater educational focus on the former, there are greater practical benefits to continuous integration of ethics early in the Computer Science curriculum. As such, I believe the integration of ethics and software development should begin early in the curriculum as opposed to the current and limited approach at the end.

I approach the problem by proposing changes to CS 2110 Software Development Methods that integrate the teaching of ethical thinking into its existing coursework.

2 Related Works

Computer Science education lacks a solid ethics educational component at most major universities or is left to the last year as optional coursework distantly related to CS. However, several major universities are attempting to continuously integrate ethical thought into their Computer Science curriculums. [3]

At Harvard University, the Computer Science and Philosophy departments are working together on a program called Embedded EthiCS. The purpose of the program is to design new introductory Computer Science courses with a greater focus on ethical thought. Embedded EthiCS attempts to embed philosophers directly into Computer Science courses to teach students how to think through the ethical and social implications of their work. Harvard Embedded EthiCS pulls in post-doctoral fellows and graduates in Computer Science and Philosophy to develop Embedded EthiCS modules. Embedded EthiCS distibutes pedagogy over stand-alone courses, exposing students to the extent to which ethical and social issues permeate virtually all areas of Computer Science. In addition, the program familiarizes students with the wide range of ethical and social issues arising across the field. Embedded EthiCS provides students with repeated practice reasoning through those issues, communicating their positions, and designing systems that consider what they have learned with five specific courses to be taken across the span of eight semesters. [4]

Another major institution pioneering the teaching of ethical thought in Computer Science is Bucknell University. Bucknell realized that many students do not get exposure to ethical thought because the teaching of it in existing courses is neatly contained in optional fourth year seminars or elective courses. The Computer Science faculty at Bucknell realized that ethical thinking in the department has a history of being valued less by curriculums than solely technical skills such as documenting code and writing clever data structures. In other words, Bucknell realized that teaching ethics as a separate course separates it as being non-technical when it is just as important that engineers are responsible for the socio-technical world. The department of Computer Science at Bucknell believes that ethics should be integrated into existing CS courses, and that ethical thinking is a habit which needs to begin practice at the same time as students begin developing their programming habits. In this way, students are able to deliberately practice ethical design throughout their career. [1]

More major universities are beginning to transition toward the integrating of ethical thinking in their Computer Science curriculums. The University of Colorado Boulder announced that its Computer Science department would be redesigning the entire curriculum to incorporate ethical design into coursework. Dr. Casey Fiesler of the University of Colorado Boulder observed that the current Computer Science coursework taught at major universities reinforces the idea ethical thinking is a non-Computer Scientists job; thus it needs to be better integrated as a part of the technical practice of Computer Science. [5]

3 Proposed Design

The ethics courses being taught at the end of the curriculum at the University of Virginia suggest that ethics is an afterthought to technical work as opposed to something that should be learned alongside it. Thus, integrating ethical thinking into introductory software development coursework addresses the minimal teaching of ethical thinking in the curriculum.

There are several changes I propose to CS 2110 Software Development Methods. CS 2110, required for all Computer Science majors, appears earlier in students' coursework—typically in the first or second year. Students in CS 2110 are also more familiar with programming than students in CS 1110. The course teaches the basics of software development and has a greater focus on building things that can be used in the real world compared to CS 1110 which teaches introductory programming skills that are not fully realized for practical application. However, CS 2110 is taken early on enough in the curriculum and because of the high enrollment, and more students such that more students may be introduced to ethical thinking in the field of computing.

Introducing ethical thinking to a broader range of students facilitates a more welcoming environment for students of different backgrounds. The improvement that I propose has the benefit of creating a positive impact on participation retention in computing from students in underrepresented groups because many ethical issues now gaining public attention are relevant to these students. [6]

Several changes could be made to the current coursework to incorporate ethical thinking. In the Fall 2019 iteration of CS 2110, students built upon a Photo Library project to learn object-oriented programming, classes, and graphical user interfaces. In conjunction with these software development tasks, a supplementary portion of the assignment could be added which provides articles on ethics and encourages students to research and think about the ethics behind social media apps like Instagram, which the assignment had been modelled after.

Another idea I propose is the addition of discussion topics which revolve around ethical issues from similar apps, such as Instagram which the assignment had been modeled after. Specific topics for the Photo Library assignment include data privacy and collection. The discussion can take the form of a short paragraph requiring students to reflect on how they would improve their algorithms and design. The discussion component can be used to integrate ideas such as privacy, UI design, and social scoring, all of which are taught in the Spring 2021 iteration of CS 3501 Everyday Ethics and Quotidian Quandaries for Computer Scientists. The discussions would reinforce the ability of students to identify and reason through ethical and social issues, communicate their reasoned position, and design ethically and socially responsible systems in the future.

Furthermore, topics from CS 3501 such as dark tactics, could be integrated into assignment instructions and submissions. Instructions can include visually separated subtasks, tables for reference on what should be returned by student's programs, and checklists for exactly student submissions. The incorporation of dark tactics develops reinforces the real-life application and emphasis on basic software developer concepts of inclusion. It conveys to students an emphasis on design creativity and consideration of multiple perspectives such as the ability of dark tactics to help student with disabilities.

Integration into CS 2110 allows ethics to become a part of future coursework in smaller doses throughout the program, as opposed to being taught once at the end of the curriculum. Doing so also reinforces the idea in students that ethics and engineering go hand in hand. [7] The changes I propose to CS 2110 do not pause the course or throw away valuable content and learning concepts, but rather resituate the existing material to involve learning about ethical design. Building habits on ethical thinking is critical to preparing students for the emerging tech landscape.

4 Conclusion

The integration of ethical thinking across Computer Science curriculums and into CS 2110 would allow students gain a better understanding of ethical issues from repeated practice and reasoning through those issues, communicating their positions, and designing ethical systems. Ethical thinking in the software development process would develop as a habit from the suggested improvements. The habit begins at the same moment students begin developing their programming habits, allowing students to deliberately practice ethical design throughout college and into their career, and reinforcing the idea in students that ethics and engineering go hand in hand. Furthermore, the process of integrating ethics into introductory software development coursework at the University of Virginia also creates a more welcoming environment for students of different backgrounds to promote diversity in a field that is actively seeking it.

5 Future Work

An important issue for consideration in future work is the trade-off between technical and ethical concepts. Introductory software development coursework integrated with ethics must ensure that students are still learning the important technical aspects of programming, such as object-oriented work taught in CS 2110. In addition, if articles and information on ethics are not incorporated seamlessly, students may become confused about the concepts that they should be learning. Furthermore, the issue of ease of course integration needs to be considered. Ease of integration would dictate that courses are not entirely overhauled, disrupting the learning of current students.

Another important aspect of future work is avoiding bias in conveying of course content on ethics. Introducing ethical topics should be done without providing specific opinions on them—just making students aware that they exist.

The integration of ethical design into introductory Computer Science coursework is not a replacement for teaching students about issues of cultural competency and identity. While the proposed changes can point to those issues in class, most Computer Scientists teaching introductory software development courses are not experts on ethics. In addition, CS 2110 at the University of Virginia has a wide diversity of students. An issue for future consideration is choosing applicable topics that speak to designing with diversity in mind that do not detract from technical concepts.

REFERENCES

[1] Evan Peck. 2017. The Ethical Engine: Integrating Ethical Design into Intro Computer Science. (July 2017). Retrieved from https://medium.com/bucknell-hci/the-ethical-engine-integratingethical-design-into-intro-to-computer-science-4f9874e756af

[2] Lou Bloomfield. UVa Course Catalog. Lou's List. Retrieved November 1, 2021 from https://louslist.org/CC/CompSci.html

[3] Paul Karoff,. 2019. Harvard works to embed ethics in computer science curriculum. The Harvard Gazette. Retrieved from https://news.harvard.edu/gazette/story/2019/01/harvard-works-to-embed-ethics-in-computer-science-curriculum/

[4] Harvard University. Embedded EthiCS. Retrieved November 1, 2021 from https://embeddedethics.seas.harvard.edu/

[5] Gernene Tan. 2020. Ethics in Tech: CU programs aim to increase ethics engagement in tech. Retrieved from https://www.colorado.edu/studentgroups/colorado-engineer/2020/04/28/ethics-tech

[6] CaSE. 2014. CaSE Report – Improving Diversity in STEM. Retrieved from https://www.sciencecampaign.org.uk/resource/ImprovingDiversityinSTEM2014.html

[7] Tom Abate. 2020. How the Computer Science Department is teaching ethics to its students. Stanford Engineering. Retrieved from https://engineering.stanford.edu/news/how-computer-science-department-teaching-ethics-its-students