Prospectus

The Integration of Ethical Design into Introductory Software Development Coursework
(Technical Topic)

Ethical Design in Introductory Software Development Coursework
(STS Topic)

By

Justin Ngo

1 November 2021

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 _________________________
Benjamin Laugelli, Department of Engineering and Society

Approved: ____ Date _________________________
Daniel Graham, Department of Computer Science

Introduction: Socio-technical Research Problem

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 (ABET, 2021). 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 (Peck, 2017). 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.

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 CS3501 Everyday Ethics and Quotidian Quandaries for Computer Scientists, and the basics of software development taught in the Fall 2019 iteration of CS2110 Software Development Methods, to propose improvements to the teaching of ethics at UVA (Bloomfield, 2021). 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. I believe that the isolated approach to teaching ethics is flawed because it does not adequately prepare students for the impactful engineering projects that 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 CS2110 Software Development Methods that integrate the teaching of ethical thinking into its existing coursework.

To better understand the benefits of synthesizing the idea of continuous integration of ethics into introductory software development coursework, I research educational literature for best practices in designing Computer Science coursework teaching the two ideas. In addition, I consider course materials from the two Computer Science courses, and materials from other outside university courses that could be implemented to improve student learning.

To redesign the coursework effectively, both the technical and social aspects of the problem must be addressed. I outline a technical process for redesigning introductory software development coursework that involves real world ethical dilemmas and increasing student engagement to introduce and raise awareness of ethical design. I also use the STS framework of Actor-Network Theory to analyze how various factors such as students, assignments, and employers have an equally important role in the facilitating the teaching of ethical design.

Technical Research Problem

As algorithms being influencing more aspects of daily life, software engineers have a greater responsibility to ensure that their creations are not causing harm. Software engineers and Computer Science students must have solid foundation in ethical design practices to prevent unintentionally harming people through their designs. As such, previous methods of teaching ethics in university Computer Science curriculums needs to be redesigned for improvement.

Current Computer Science coursework at the University of Virginia 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: STS1500, STS4500, and STS4600. STS4500 and STS4600 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 CS3501 Everyday Ethics and Quotidian Quandaries for Computer Scientists, or later courses that contain limited integration of ethics such as in CS3240 Advanced Software Development (Bloomfield, 2021). These courses are optional and not required for Computer Science students in the College of Arts and Sciences. In addition, these courses are also not intended to be taught in the first two years of a student's Computer Science coursework.

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. Therefore, integrating ethical thinking into introductory software development coursework addresses the minimal teaching of ethical thinking in the curriculum. As such, I propose several changes to CS2110 Software Development Methods. I also believe

CS2110 should be the subject of redesign for several reasons. CS2110, required for all Computer Science majors, appears earlier in students' coursework— typically in the first or second year. Students in CS2110 are also more familiar with programming than students in CS1110. CS2110 also teaches the basics of software development and has a greater focus on building things that can be used in the real world compared to CS1110 which teaches introductory programming skills which are not fully realized for practical application. However, CS2110 is taken early enough in the curriculum, and because of the high enough enrollment, allows for-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 added 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.

In the Fall 2019 iteration of CS2110, students built a Photo Library project to learn object-oriented programming, classes, and graphical user interfaces. The course would benefit from the addition of discussion topics revolving around ethical issues from similar apps, such as Instagram, which the assignment had been modeled after. Specific discussion topics for the Photo Library assignment could include data privacy and collection. The discussion could take place in the form of a short paragraph requiring students to reflect on how they could ethically 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 CS3501 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.

Integration into CS2110 allows ethics to become part of future coursework in smaller doses as opposed to once at the end of the curriculum, 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. The changes I propose to CS2110 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 (Peck, 2017).

STS Research Problem

Despite the growing need for stronger ethical design in the field of Computer Science, 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 Computer Science. However, several major universities are attempting to continuously integrate ethical thought into their Computer Science curriculums (Karoff, 2019).

A 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 Computer Science courses, and that ethical thinking is a habit which needs to begin as soon as students begin developing their programming habits. In this way, students can deliberately practice ethical design throughout their career (Peck, 2017). The program has gained attention and praise in the field, and is inspiring the development and integration of similar ethical reflection coursework at other universities (Tan, 2020).

Actor-Network Theory (ANT) considers human and non-human elements as equally weighted actors within a heterogenous network (Cressman, 2009). Student assignments and coursework designed and taught by professors heavily influences student practices and diversity within the field. Student practice of ethical design throughout university coursework is important to designing with ethics in mind in ~~their professional practice~~ the field. In addition, company involvement in valuing ethical design as well as seeking students with ethical thinking skills encourages student participation in learning ethical design. Furthermore, there is a positive impact on participation retention in computing from students in underrepresented groups because many big ethical issues are relevant to these students (CaSE, 2014). Retaining greater diversity within the field reinforces better ethical design practices with underrepresented groups in mind.

Ethics courses being taught at the end suggests it's an afterthought to the technical courses as opposed to something that should be learned alongside it. Thus, early courses and introductory programs can help address these flaws (Abate, 2020). Integration should happen in smaller doses throughout the curriculum as opposed to once at the end. Doing so also reinforces the idea in students that ethics and engineering go hand in hand. Under ANT, the social, technical, and political events are equally important to ensuring ethical design in professional practice. If any actors and improvements are left out, then the successful ethical design cannot be achieved.

Conclusion

The integration of ethical thinking across Computer Science curriculums and into CS2110 would allow students to 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.

The process of integrating ethics into introductory software development coursework at the University of Virginia 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 the field. In addition, it develops the practice in students of considering ethics not only in college but throughout their career in Computer Science.

Word Count: 1889

References

Abate, T. (2020, August 20). How the Computer Science Department is teaching ethics to its students. Stanford Engineering. https://engineering.stanford.edu/news/how-computer-science-department-teaching-ethics-its-students

ABET (2021). Criteria for Accrediting Applied and Natural Science Programs, 2021-2022 https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-applied-and-natural-science-programs-2021-2022/

ACM (2021, November 1). ACM Code of Ethics and Professional Conduct https://www.acm.org/code-of-ethics

Bloomfield, L. (2021, November 1). UVa Course Catalog. Lou's List. https://louslist.org/CC/CompSci.html

CaSE (2014, May 14) CaSE Report – Improving Diversity in STEM. https://www.sciencecampaign.org.uk/resource/ImprovingDiversityinSTEM2014.html

Cressman, D. (2009, April). A Brief Overview of Actor-Network Theory: Punctualization, Heterogeneous Engineering & Translation.

Harvard University. (2021, November 1). Embedded EthiCS. https://embeddedethics.seas.harvard.edu/

Karoff, P. (2019, January 28). Harvard works to embed ethics in computer science curriculum. The Harvard Gazette. https://news.harvard.edu/gazette/story/2019/01/harvard-works-to-embed-ethics-in-computer-science-curriculum/

Peck, E. (2017). The Ethical Engine: Integrating Ethical Design into Intro Computer Science. https://medium.com/bucknell-hci/the-ethical-engine-integratingethical-design-into-intro-to-computer-science-4f9874e756af

Tan, G. (2020, April 20). Ethics in Tech: CU programs aim to increase ethics engagement in tech. https://www.colorado.edu/studentgroups/colorado-engineer/2020/04/28/ethics-tech