

Case Study Based Pedagogical Intervention for Teaching Software Engineering Ethics

Aditya Johri
Information Sciences & Technology
George Mason University
Fairfax, VA USA
johri@gmu.edu

Ashish Hingle
Information Sciences & Technology
George Mason University
Fairfax, VA USA
ahingle2@gmu.edu

Abstract—The omnipresence of software systems across all aspects of society has necessitated that future technology professionals are aware of ethical concerns raised by the design and development of software and are trained to minimize harm by undertaking responsible engineering. This need has become even more urgent with artificial intelligence (AI) driven software deployment. In this paper we present a study of an interactive pedagogical intervention – role-play case studies – designed to teach undergraduate technology students about ethics with a focus on software systems. Drawing on the *situated learning* perspective from the Learning Sciences, we created case studies, associated stakeholder roles, discussion scripts, and pre and post discussion assignments to guide students’ learning. Open-ended data was collected from thirty-nine students and analyzed qualitatively. Findings from the study show that by taking on different perspectives on a problem, students were able to identify a range of ethical issues and understand the role of the software system process holistically, taking context, complexity, and trade-offs into account. In their discussion and reflections, students deliberated the role of software in society and the role of humans in automation. The curricula, including case studies, are publicly available for implementation.

Keywords—software engineering ethics, ethics education, case studies, role plays, situated learning

I. INTRODUCTION

Whether it is self-driving cars, recommendation systems for shopping, or applying for a loan, the impact of software on people and society is now inescapable [1-2]. As software gets increasingly embedded across technological artifacts in society, the need to better understand its intended, unintended, anticipated, and unanticipated impacts becomes more acute. The increased use of complex algorithms, machine learning, and the advent of artificial intelligence (AI), which are harder to understand but even more impactful, intensify the need for ethical and responsible software development [3-4]. The field of software engineering ethics, which examines these issues, has thus become increasingly critical within software engineering (SE). Yet, within software engineering education, the topic of ethical responsible design, development, and deployment is neglected and needs to be paid more attention. Given the importance of ethics to software engineering, other scholars have also advocated for a stronger position within SE on ethical education [5] that “takes into account both rules and consequences to assess the goodness of actions, and at the same

time pays an adequate consideration to the absolute values of human dignity” [6, pg. 505].

Over the years, software engineering ethics has primarily focused on two aspects of the process: the choices made by engineers or developers and the professional codes that guide their practices and software development. A focus on professionals in the field is essential, as they design and develop the systems that are implemented across industries and organizations. Since the user facing software engineering work is largely done in the industry, unsurprisingly, the focus on ethics has been on professionals. We argue that it is equally important to focus on students or future software engineers and their training related to software engineering ethics. If students are exposed to ethical concerns early on, they can learn to think holistically about responsible development and use of software. Higher education is also a time in their education and career when they have time and support to delve more into ethics.

Many scholars have recently developed ways to teach ethics to provide stronger contextual training to students [7-8]. In this paper, we present a research study of a pedagogical intervention designed to teach students the ethics of software in an interactive manner through the use of role-play case studies. Role-play case studies are a form of simulation exercise or training that provides students with a real or fictional scenario and narrative, assigns them roles, and asks them to deliberate and discuss a possible solution [9]. The ability to make ethical decisions relies to a large extent on the ability of the decision-maker to take context into account to understand not just the immediate technical need of the work but also larger implications that might even result from unanticipated consequences. We designed multiple case studies related to the design and use of software that were conducive to role-play. We implemented them in a class for information technology students that focused on teaching the social and ethical implications of technology. We collected data using multiple sources – survey, focus group, discussion transcript – and analyzed them iteratively to understand student learning outcomes. Overall, we found that our intervention and approach led to a measurable change in students’ understanding of the ethics of software.

In the rest of the paper, we first review prior work on software engineering ethics, situated learning – the theoretical perspective underpinning our approach, and the use of case studies in ethics education. This is followed by a discussion of the design of role-play case studies and the context of

implementation, including course and student details. Research study and findings follow this, and we end with a discussion and conclusion section. Our intervention has so far been implemented across ten class sections over four years (see [43-45] for details), but for the purposes of this paper, we limit the research study to multiple cases over one course section to provide a comprehensive and in-depth assessment of learning outcomes.

II. PRIOR WORK

A. Software Engineering Ethics

Software engineering ethics is a multifaceted field that examines several interrelated approaches to the ethical development of software. Gotterbarn in particular, has made significant contributions to the field starting with the software code of ethics advanced by ACM in 1997 [10-12]. It emphasizes understanding the practical decision-making of software engineers as well as the creation of guidelines or codes that guide software development [13-15]. The field is founded on the principle that technical decisions and human values are intertwined and software engineers have can make a “significant positive impact on society, simply by becoming more sensitive to the long-term human relations implications of our work and incorporating this sensitivity into our software designs and products” [16].

Two recent works within the field are of particular relevance to the pedagogical approach and subsequent assessment reported here. First, an applicable framework that has been advanced is the Ethically-Driven Software Design (EDSD) [17]. EDSD is a practical approach to software development that creates a link between ethical and professional skills making the overall process more transparent and also brings in an awareness of risks and limitations involved. The application of this framework is through the use of specific index cards to aid the design of software [17]. These cards prompt responding to an ethical question from the viewpoint of different stakeholders involved with the software design process including customer, manager, engineer, etc. The design and use of these cards aligns with the role-play-based intervention we have designed. In both cases, the ability to take a perspective and offer a viewpoint is central to the design or learning process. The need to use or play a role ensures that multiple stakeholder values are represented. This ability is crucial to both documenting and addressing ethical concerns.

Second, in their roadmap for ethics-aware software engineering, Aydemir and Dalpiaz [18] propose an analytical framework that assists stakeholders in analyzing ethical issues in terms of subject (software artifact or SE process), relevant value (diversity, privacy, autonomy, etc.), and threatened object or stakeholder (user, developer, etc.). More critically, from the perspective of the current study, they identify a range of ethical concerns that are relevant for pedagogy including transparency, awareness, privacy, accountability, diversity, dependability, and the role of business and work ethics within the software development process. Overall, the elements they identify align closely with other similar ethical and responsible software development aspects that have been advanced, especially in relation to algorithmic and AI-driven systems. For the purposes of this study, their list provides a mechanism to analyze the data

to identify ethical aspects that students bring up in their discussions and responses.

Overall, prior work on software engineering ethics emphasizes examining software engineering and the implications of software use from a more contextual perspective considering viewpoints of different stakeholders. Similar to other ethical frameworks, it reiterates taking responsibility for developing software and doing no harm [19-21].

B. Situated Learning Perspective

One of the theories of learning that allows us to design and assess pedagogy that focuses on teaching paying attention to contextual aspects of an issue is situated learning [22-24]. The situative perspective emphasizes the role of context on scaffolding an individual’s conception of knowing and how they learn; according to situative learning, knowledge is not something that an individual possesses or stores in the brain but is present in all that they do [25]. Consequently, people learn best through working on socially relevant problems in collaboration with others.

The idea within the situative perspective that is particularly relevant to the proposed work is that what we learn and how we learn it is closely linked [26]. This situation, or context, determines what people look for, how they get the information, and how they interpret it. Knowledge arises dynamically and is constantly constructed and reinterpreted. Therefore, both in terms of how to teach and what to teach, this viewpoint highlights the importance of context and social practices. Within the situated learning paradigm, Greeno and van de Sande [27] further advance a perspectival view which argues that a person’s or group’s knowledge and understanding of any conception is “their ability to construct perspectival understandings that are situated in activity and that are organized according to principles that are taken as defining the conception (pg. 14)”. This motivated our use of case studies and, specifically, role-plays as pedagogical tools that can support ethical thinking among students, and place ethics more centrally within the course and curriculum [28].

C. Case Based Ethics Education

Ethics education has a long history within higher education [29]. Case studies are short narratives that condense a real-world problem or concern and can be focused on a narrow problem or a larger-scale project [30-31]. They are commonly used for teaching ethics within engineering and computing and employed both to teach how to apply professional codes or even broader societal considerations of engineering. The use of case studies for education originated in business schools and continues to be a popular form of teaching in that field. [30] have proposed that case studies are also good for teaching software engineering. Case studies and, in particular, role-play-based discussions are effective because, as prior research on situated learning has demonstrated, there is value in inculcating perspectival thinking and that case studies and collaborative discussions are effective in achieving this learning outcome.

Role-playing, a specific type of case discussion informed by simulation-based pedagogy, involves using cases or scenarios to discuss a problem or an issue where each participant is assigned a specific role [9, 33, 34]. Within the context of teaching,

students take on the role of a person who is involved with or impacted by the problem outlined in the case, and they have to respond to questions and take part in a discussion in a manner that reflects the viewpoint or perspective of the role they are assigned. Role-play supports learning across a range of areas, including perspective-taking, critical thinking, and communication skills [35-36]. Furthermore, a role-play-based approach lends itself to a broader discussion around the role of technology from multiple perspectives.

Within software development and software engineering, in addition to the actual algorithms, programming environments, and organizational details, there is the need to better understand the use of a specific software system at the point of implementation. The usefulness and subsequent continued use depend on how users are able to integrate a system in their work and life practices. This also implies that to reduce any unanticipated outcomes that can be harmful, it is important for an engineer to take the overall context into account. This is where a focus on ethics is important, and the use of pedagogical strategies that teach students how to build a mindset that supports their intention for ethical and responsible development. Within software engineering pedagogical practices such as role-based group work, scenarios, ethical deliberation, and social engaged learning are already being used in some form, and our approach is an adaptation and extension of this work [37-40].

III. CASE DEVELOPMENT AND IMPLEMENTATION CONTEXT

We leveraged prior work on the design of role-play case studies to create four software-related cases. Each case included a real-life inspired scenario encapsulating a software-driven problem within a larger context.

1) *Boeing Case*: This case focuses on the Boeing 737 Max aircraft crashes, specifically, the use and implementation of the Maneuvering Characteristics Augmentation System or MCAS software and Boeing's failure to inform the pilot about MCAS.

2) *Volkswagen Emissions Case*: This case focuses on the use of a software system by Volkswagen to cheat on emissions testing for its diesel engine vehicles in the U.S. and the aftermath of that scandal.

3) *Facial Recognition Case*: The case focuses on the use of facial recognition by a university to track people on campus to ensure they are fully compliant with post-Covid requirements.

4) *Credit and Lending Case*: This case focuses on the use of novel forms of data and models by a bank to make credit and lending decisions.

For each case, we created at least six to eight different roles, including a moderator role - a stakeholder who would lead the discussion and engage all the other participants. Each case also included relevant readings and video resources. Some resources focused on specific use cases of technology, while others were broader and introduced ethical frameworks or ideas. Students were encouraged to undertake additional research to prepare for their role, and most reported doing so. Table 1 summarizes the case, participant roles, and ethical elements the case targeted. *Additional information about the cases, including the full text, is available online publicly and will be added post-review.*

We implemented the cases across ten class sections with 30-45 students each between 2020 – 2024. The students were divided into groups of 5-7 for the role-play discussions, and these groups remained largely consistent. In each course offering, at least four cases were implemented. We used a design-based research approach to understand the implementation outcome and evaluate the intervention (anderson2012design}). Based on student feedback and our observations, we systematically revised aspects of the cases and their implementation in different iterations. For instance, we added more background reading about technical concepts across all the cases, created more roles for larger sections, and recorded a mock video demonstrating how to run a role-play.

TABLE I. CASE DETAILS

<i>Case</i>	<i>Description</i>	<i>Roles</i>	<i>Ethical Software Concerns in the Case</i>
Boeing	As a member of the Aviation Transportation Investigative Committee (ATIC), participants are tasked with discussing how systematic (technical and social) issues in aviation can be mitigated.	Aviation Consultant Aerospace Eng. Professor Software Eng. Retired FAA Officer Boeing Representative Family Lawyer	Accountability Transparency Business concerns Dependability Work concerns Safety
Volkswagen	A rental company's fleet manager is interested in VW vehicles. They want to know if the company can be trusted and what responsibility organizations have to correct the problems they cause.	Consumer Adv. Group Chair Env. Engineering Professor Env. Advisory Group Director Software Developer Automotive Industry Expert Compliance Director	Trustworthiness Sustainability Business concerns Work concerns Accountability
Facial Recognition	In response to the global pandemic, Andrew Hamilton University administrators have established a task force to discuss if facial recognition technology will ensure students' safe return on campus.	V.P. of Campus IT V.P. of Student Government History Professor A.V.P. in Provost's Office Equity and Inclusion Director FRT Non-Profit Director	Privacy Surveillance Diversity Transparency Accountability
Credit and Lending	A loan manager wants to explore algorithmic decision-making for credit and loan decision-making and whether data on demographic and socio-economic profiles should be included.	Data Analyst Underwriting Specialist Chief Credit Innovation Officer Financial Research Analyst Loan Process Regulatory Officer AI Ethics Adv. Group Director	Accountability Transparency Bias Diversity Privacy

B. Course Details

The course, consistent across the four years of this work, was an undergraduate offering in IT designed to teach students about the role of technology in society. The course fulfills the ABET accreditation related to ethical and professional responsibilities for students in degree programs, including cybersecurity, information technology, and data analytics (broadly described as technology students). The students in the program take programming and software development as a required course. They also undertake software development in their capstone course either concurrently or after this course. Most students graduate to enter the IT industry as programmers/software developers.

The course itself was divided into 2 - 3 week long modules. In the first week, students were introduced to a topic through papers and videos and provided the case study and assigned roles. In the second week, students were given pre-discussion assignments, which included answering a set of questions from the perspective of their role and drawing a concept map on their recognition of broader concepts. In the third week, students would participate in the role-play discussion activity during the class session. In some cases, weeks 2 and 3 were combined. After the session, students would complete a post-role-play assignment and a collaborative group concept map. In some instances, they held a debrief discussion rather than drawing a concept map. Table II lists the steps for role-play application.

TABLE II. APPLICATION OF ROLE-PLAY IN CLASS

1	Introduction to module (topic)
2	Videos covering module material
3	Readings covering module(includes news & scholarly articles)
4	Distribute the role-play scenarios and assign roles
5	Pre-discussion questions
6	Role-play activity in class
7	Post-discussion questions
8	Debrief and discussion

IV. RESEARCH STUDY

The overall study was designed as a mixed-methods approach with a mix of data. We created and used different assessments for student learning, including open-ended questions given to students through pre-and-post assignments and concept maps to map the expansion of student concept recognition. We also used focus groups and end-of-semester evaluation data to assess the efficacy of the role-lays. This research was approved by the Institutional Review Board (IRB), and only data from students who consented to the study was analyzed. To protect the confidentiality of participants, data analysis took place after the final grades were assigned, and all data was anonymized by a third party not involved with grading.

Forty-one (41) undergraduate students participated in the course, of which thirty-nine (39) consented to the study and engaged with the role-play case studies. The students' degree programs varied, but all students were required to take at least one software development course before or concurrently with this course where data was collected.

A. Data Collection

For this study, the data used for analysis included pre-discussion assignments (completed before the role-play activity), post-discussion assignments (completed after the role-play activity), and end of the semester assignments.

The pre-discussion assignment included the following questions: What role are you playing in the role-play group discussion? From the perspective of your role, what will be your recommendation? From the perspective of your role, how can we ensure future safety and transparency and rebuild trust? Why do you think the approach you suggest is the best approach? What do you think are the main barriers to this approach? The post-discussion assignment asked the following questions: What response was reached following the discussion and what criteria were considered to reach this solution? Was the response agreed to by all or did one person have more influence? Why? Do you personally agree with the response? Why/Why not? Any comments on how your group approached the case? There were minor variations in these questions based on the specific case.

The final assignment asked them what they learned about ethics from the course and also individually from each case study. Finally, the primary author moderated the role-play sessions and was cognizant of the discussions that took place. Data was curated by the second author.

B. Data Analysis

Analysis was done iteratively by both authors. Since in this particular study we were interested in understanding student learning using open-ended responses, a qualitative research approach was deemed appropriate [41]. Consistent with qualitative data analysis, data were open-coded, and then specific items, such as ethical principles and connections to the cases were identified in the data [42]. Once the authors were familiar with the data, they read through and tagged the student responses for common ideas consistently brought up in the discussion. Table III highlights these common ideas as "initial codes", specifically how often the aspects of ethical and responsible software were discussed through the dataset. These initial codes were explored in more detail through the findings. Both authors did multiple round of coding to ensure consistency and reliability, and to select examples for this paper.

V. FINDINGS

The purpose of the course and the case studies was to introduce students to different aspects of ethical and responsible software engineering. Consequently, cases touched upon different aspects of ethics with some overlap (see Tables I & III). In this section, we present findings grouped first by the different aspects of ethics and then from a more situated perspective to capture students' understanding of the complexity of the

engineering process. To support and illustrate examples, quotes from the students are included. The quotes were minimally edited for anonymity and editing, but care was given not to alter meaning. Student numbers in [square brackets] used with the quotes refer to their corresponding deidentified number in our dataset.

TABLE III. ETHICAL CONCERNS INSTANCES

<i>Initial Codes</i>	<i>Boeing</i>	<i>Volkswagen</i>	<i>Facial Recognition</i>	<i>Credit and Lending</i>
Accountability	7	5	4	6
Trust /Dependability	6	8	3	2
Bias	0	1	7	22
Diversity	0	0	10	1
Privacy	0	0	20	4
Safety	8	3	10	0
Sustainability	0	18	1	1
Surveillance	0	1	25	0
Transparency	5	6	3	11
Business Concerns	15	6	0	9
Work Concerns	17	4	1	5

A. Aspects of Ethical and Responsible Software Discussed

a) *Accountability*: Different stakeholders are involved in any system development or software engineering project. Therefore, the accountability in the case of an error or fault is an important element to identify. Through the case studies, students were able to discuss the issue of accountability and through the Volkswagen case study, they learned that many companies do take accountability and make amends for the harms they have caused.

“Volkswagen’s response to the scandal [was] to blame the software engineers, or ‘rogue’ software engineers. The blamed software engineers solely for programming the ECU to make the engine go into a weaker mode... as if the software engineers would care at all to implement it in the first place. As if Volkswagen had no testing from the hardware aspect, or no actual QA or testing of the cars. It took years for this to be discovered... where eleven million cars rolled out of the factory like this, and they completely push the blame off themselves and just blame it on the software engineers who worked there - as if they get their kick out of reprogramming these cars for company profit that they’ll never see.” [Student 12]

“Boeing was not as transparent with their problems as they could have been. It seemed that they tried to blame other factors outside of their plane/product before admitting there was a problem.” [Student 1]

b) *Trustworthiness/Dependability*: Dependability refers to the degree to which an organization or a person can be counted on to fulfill their responsibilities, including ensuring that systems work as intended and that users are fully aware of the outcomes of their actions while using a system. Dependability, in some ways, is related to the factor of trust as being able to depend on someone or something for the long term generates trust. Whereas accountability focuses on taking responsibility for actions, dependability focuses on a consistent attitude of trust. This is a complex element as different components of a system have to pass reliable information:

“This disaster could have been prevented if, instead of continuing to create more and more complex planes, we stuck to a simpler design and just streamlined it from there. Modern planes have become too complex and have too many bells and whistles. By designing the plane to need auto-corrective software, we made it so that the planes would essentially try and fly themselves based on various different sensors. Depending on what those sensors detect, the plane could either have a smooth and safe trip, or plummet into the ground based on false information from the sensors. By having the new Boeing Max planes be reliant on the software that auto-corrects the angle of the plane and pushes the nose down, you have essentially put another sensor into the mix that could potentially be fatal. If we had designed the plane in a way that would have not needed the use of this software, or at least been transparent about what software had been newly installed, then most likely this disaster would have never happened.” [Student 6]

“In order to ensure future safety, transparency, and trust, the solution is for Boeing to slow down on its desire to rush its design of aircraft, not pressuring the FAA to speed up regulations, and for the FAA to have more funding. If Boeing slowed down its design process, they could create solutions to any problems it faces, instead of temporary solutions such as the MCAS system. This would in return ensure more safety.” [Student 9]

c) *Bias*: The problem of bias is becoming commonplace as software – algorithms – is used to make decisions that impact people directly. Based on datasets that themselves can be biased, software-based decision-making can affect people negatively, and these concerns are present in the facial recognition, credit and lending cases. Through the role-play discussions and the readings, students learned about the issue of bias and also debated ways to mitigate the problems.

“From an ethical standpoint, I learned that bias occurs frequently and that when it is involved in real-life situations such as credit lending or court cases, we should do our best to mitigate it. [The lending case] showed me the importance of bias which I will always take into account when AI based credit lending becomes the norm.” [Student 21]

d) *Diversity*: The ability to include a diverse population and be inclusive in terms of the workforce and the user population, is critical for development of software that serves everyone. If addressed correctly, diverse representation can address bias to some extent. Consequently, diversity is a core

ethical concern for software engineering. This issue was central to the facial recognition and lending and credit cases and students identified and discussed them:

"For the Lending/Credit/Loan case study, I learned that there should not be any biases within the algorithm that can influence the application's decision. Also, alternative data as a factor can be unethical, such as the application being denied based on the application's demographic and socio-economic. According to my previous research, while most banks have regulation and transparency on how they approach their decision for every application, they lack diversity which I believe is unethical. Everyone deserves the same chance when financial institutions are deciding on loan applications." [Student 39]

"Facial recognition (FR), in its current form, cannot be used on a diverse racial and economic student body. FR systems have inherent bias stemming from the training sets used by many companies. A recent study, conducted by the National Institute of Standards and Technology (NIST), found that facial recognition algorithms were between 10 and 100 times more likely to misidentify Asian and African Americans. These stark differences in FRs accuracy show how primitive and dangerous implementing this technology could be. Imagine a scenario where minorities at the university are regularly harassed by our security, despite completing the health check, because of the FRs inaccuracies. We must create a safe environment for students and faculty. However, we must also work to create an environment that does not further entrench racism and bias in our community. The current system proposed not only discriminates with its use of FR but also its reliance on smartphones. With COVID-19 leaving many students and their families without a stable income, it is doubly immoral for the university to assume access to these devices." [Student 18]

e) **Privacy:** The impact on privacy of new forms of software is one of the most recognized ethical impacts and students brought it up during the facial recognition discussion.

"There are benefits to using facial recognition on university grounds; however, it does not outweigh the consequences. Our top priority should be protecting the university's students' safety and privacy during COVID and after. Using facial recognition at AHU will not protect the students' safety or privacy, especially our students of color. Instead of incorporating a biased system that will disproportionately affect students of color, we should carefully follow and enforce CDC guidelines on campus as we have been the past year." [Student 4]

f) **Safety:** Safety was often a critical element of the cases, in particular the Boeing case where the disaster resulted in very high number of casualties.

"While learning about the Boeing incident, I understood how a seemingly minor issue (such as competing with another company for sales), could potentially put others at risk. Boeing is a prime example of why you shouldn't skim on quality or time when it comes to something as

serious as the lives of 350 people. The lessons on ethics learned with this scenario are to always be honest and transparent with others and to not prioritize financial gain over the safety of others." [Student 27]

Safety was also present in another form where the use of technology would make the users safe, e.g., in the case of using facial recognition on campus. For students, the issue of safety was interlinked with other elements, and they saw it both as a non-debatable element, but also as a trade-off.

"We came to the conclusion that everyone valued and needed their privacy. For this reason, we were against the idea of using facial recognition. However, we recognized that something must be done to ensure the safety of the college. So, we came up with an alternative solution, which was to examine everybody directly instead of using technology that would only violate our privacy." [Student 26]

g) **Sustainability:** Environmental aspects of information technology are increasingly coming into focus with the climate crisis and sustainability is not considered an integral aspect of ethical and responsible software engineering. The Volkswagen case study brought this issue into focus for the students.

"The Volkswagen case study showed me that even the most respected companies are capable of behaving unethically. Before this scandal, Volkswagen was one of the most well-respected car companies. Cheating car emissions tests is completely unethical, and extremely bad for the environment. This case study shows me that we need to hold companies to an ethical code, to ensure that they don't do any unethical business practices." [Student 28]

The students realized that the effect on the environment of software use does not necessarily have to be direct but indirectly, software can play a central role by impacting how a system is checked or verified.

"Because my role wasn't very ethical, I could view the situation from two completely different perspectives. I'd say the main lesson on ethics in this scenario is that even if you aren't necessarily putting people in harm's way (like Boeing, for instance), deceiving customers is still unethical. Additionally, Volkswagen was still harming the environment. I think being open and honest with others is important, especially when running a business." [Student 27]

"In the Volkswagen scenario, I learned that no matter what I try to do to make my products seem better, it is unethical to try and cheat the system when it affects the environment. No matter if a company is trying to make a few extra dollars or they are trying to make a tight deadline, installing a device that purposefully changes the emissions to meet the standards, and then after the inspection to raise the emissions to make the vehicle output more power, is not the ethical thing to do in any situation. If it was on a smaller scale then I believe that VW would not have gotten the flak that it did, but this emissions issue affected vehicles across the globe." [Student 6]

h) *Surveillance*: The ability to surveil people has increased dramatically with new technologies, including digital cameras and software that can process the captured video. Especially in terms of surveillance through facial recognition, there is a need for increased awareness, and through the case study related to the use of facial recognition on a college campus, students both became aware of this topic and discussed it from different viewpoints.

"The main criteria when choosing alternative options to facial recognition was really that we are opposed to the idea of surveillance and our data potentially being sold to third parties we are unaware of. This is why we agreed on COVID testing and temperature checking over facial recognition." [Student 27]

In addition to the proposed software implementation, students also discussed their personal experience with the technology, including their lack of understanding of how some of this software works and the need for more transparency.

"With privacy and facial recognition, students are just not going to be aware of the extent this technology will be used. We're already concerned about facial recognition tracking software for our exams, so if we have constant surveillance that impedes our privacy like this, it will just not be very good for the mental health and well-being of students. The other issue is the bias and discriminatory problems that could inhibit the facial recognition technology. We don't know if the AI will target certain demographics, like people of color, so it's a questionable tech." [Student 12]

i) *Transparency*: Being communicative and clear about a process or why a change is being made leads to people outside of the business being able to perceive what is happening with a software or process. Transparency in software development does not mean total visibility of the internal components. Transparency can be encouraged between the business and users, developers and users, or among other stakeholders. Especially when navigating business concerns, the cases highlight transparency's ability to highlight other business values.

"Boeing was not transparent with its pilots or the public. There was little to no communication within the company and it resulted in a lot of lives being lost. The public lost a lot of trust in this company due to the lack of transparency." [Student 25]

"The best approach is to make the AI, but make it public, or be able to show it to the people you are using it for. With discrimination and bias it can be bad, but Kwame is in the middle. He knows the ins and outs of AI and would make it so that it does not have an ounce of bias in it, not with demographics or skin color. He would work to make it transparent, which is what it needs." [Student 33]

j) *Business concerns*: Business activities often raise ethical questions or concerns that, if not addressed, can result in issues escalating. Engaging with the cases and the discussion that follows raises many of these concerns and highlights the

sometimes opposing values of business activities that result from a decision.

"From my perspective as Mary Bradley, I would say out of experience and for the time I spent in the company, the disaster happened due to the competition between them and Airbus, and how the build of the airplanes was rushed and didn't provide the safety needed. It could've been prevented if they gave themselves more time to build this project and be more careful and safer and not worry about the competition." [Student 10]

k) *Work concerns*: Organizational culture often determines how employees act and what values they prioritize and it is another important aspect of ethics that students need to be aware of, and learned through their discussion:

"The company culture at Boeing proved to be their downfall. A breakdown in collaboration and business executives caused over 300 people to die. Greed motivated the company to cut corners and prioritize good business over good engineering...The VW role-play, like the Boeing role-play, shows how a breakdown in communication can lead to compromised ethics within an organization. Like Boeing, VW business executives were unwilling to listen to their engineers and as a result, cut corners. These cut corners destroyed their reputation and the environment." [Student 18]

B. Situated Factors

a) *Complexity and Contradiction*: As highlighted in the discussion of the aspects of ethical and responsible software discussed in Table 1, many of these issues occur concurrently. This leads to situations where there is a lot of information to process for a decision to be made addressing the level of complexity in the case.

"All members of my group realized that the Boeing Max disaster wasn't the fault of a single person, but rather a combination of events by all parties involved. The group also realized that moving forward, the only solution is for everyone involved to be more open to each other and be willing to work with each other." [Student 9]

"The ethical oversights made by Volkswagen include company culture and transparency. Their company culture was a toxic work environment where engineers did not feel comfortable bringing up their concerns to management. Management promoted a mindset where it must get done no matter what, and if you cannot do it, someone else will. They also lacked transparency to the consumers and the EPA on the emissions that this car produced, which caused more harm to the environment. Their response to the crisis was handled quite well. They issued refunds for all cars with the problem, the CEO resigned, and they continued with transparency through the entire process." [Student 1]

There is also room for contradiction, as the roles encourage students to think about the cases in different ways, and students will often come up with many smaller micro-decisions leading toward a larger decision about the whole case. Navigating the values laden in these micro-decisions outlines the contradictions as they navigate the ideas.

“Volkswagen failed to demonstrate a strong corporate social responsibility. They ignored emissions regulations earn higher profits. The emissions rates were damaging to the environment and people, who could develop cancer from the smog produced. I have a mixed opinion of Volkswagen’s response to the crisis. One thing that I think was done poorly was the lack of accountability. Volkswagen committed to transparency, however, continued conducting shady practices such as installing hidden devices in their cars. One thing that I think was done well is Volkswagen replaced their CEO. The new replacement promised deep reforms and a focus on developing electric cars.” [Student 24]

“It taught me that we do not always think about how technology will affect average people and how it will benefit the people implementing them. The facial recognition technology had racial bias and other factors that would negatively effect people on the campus, but it also seemed to have lots of benefits it would provide to the university.” [Student 17]

b) *Human Factor in Automation:* Students recognized that although the use of technology in many cases was inevitable given its potential usefulness, they did realize that to ensure that users were not harmed unnecessarily or unintendedly, a human-in-the-loop approach was needed. Algorithm-driven systems need to work in conjunction with humans.

“I think we need to work with decision-making algorithms side by side. The modern decision-making algorithms are good but they need human supervision. Their decisions can’t be trusted completely. The final decision of these algorithms needs to be double checked by humans working on the same project. Hence, in the present time I think the best approach to make a decision on the loan application is by working with both algorithms and humans. I don’t see any barriers to this approach if we work both with the algorithms and humans combined.” [Student 30]

c) *Limits on Technology Use:* Finally, for some students, the ethical decision was not to use technology at all, especially if there was no way to control some of the factors and downsides associated with the use of a system:

“My group came to the conclusion that using advanced technology is not always the solution. There must be a purpose and a real need to use it, otherwise it would cause more harm than good. It is important for people to understand what this technology brings to the table and what the pros/cons are of using it. While using FR might be useful and a good way to keep the students on top of the COVID-19 situation, it is only a short-term solution. What would this technology be used for after the fact, or where will all the stored data go? These questions must be asked by everyone because it is ultimately up to us to have a say in what we do as a community on campus.” [Student 22]

C. Overall Learning Related to Ethics

Finally, we report overall learning related to ethics among students. For many students, this was the only course related to ethics they took and the only time they were exposed to case

studies. Therefore, both the topic and the pedagogical intervention were new to them.

“Before taking this class, I never truly gave ethics a lot of thought. I have deep-rooted moral principles that I have always followed, but have never truly related those principles to the real world environment, specifically in a professional environment. After taking this class, you have opened the door to deep thought about ethics in the professional world. I have discovered new principles that will help me later in my career.” [Student 23]

Most students would go on to have a career in the information technology industry, starting at entry-level and working on and with different software.

“As I am graduating in a couple of weeks, this course was one of my favorite courses I have taken at [the university]. As a vegan and intersectional activist, ethics have always been really important to me. In my years studying IT, it did not feel fulfilling because it felt like I wasn’t doing anything impactful for society. However, after this course, and especially during the lending/loan topic, I feel inspired to work in a field and create things that prioritize doing good for society while still being able to pursue IT.” [Student 4]

“I learned that AI can be biased, which made no sense to me at first. I found it crazy that things like AI could have bias, especially since it is literally a computer system. I also didn’t realize that there was a certain type of bias regarding loans and lending. I know that credit history and all that matter when pulling a loan, but to see a computer deny people was surprising. Hopefully, in the future, AI won’t dominate the world.” [Student 33]

For many, the first tasks would be programming-related. Consequently, the topic of software development was of inherent interest to them.

“After learning the extent to which Volkswagen went to cheat emissions tests, I learned that all companies, especially tech, have the ability to cheat or alter the way their software or information works because they are the creators of this sophisticated technology that we have zero knowledge of on the inside. For example, VW hired skilled programmers to implement complex code or Apple secretly underclocked or slowed down iPhones to fix battery issues. It’s something to consider, especially with new and upcoming advancements.” [Student 21]

Interestingly, preparing for their roles motivated students to undertake extra research and learn more about the topic beyond what was needed for the case discussion, often leading to strong insights about the need for learning about ethics in relation to software systems.

“The facial recognition role play sent me down a rabbit hole of articles. Researching this, I found out how much other countries invade their citizens’ privacy. These countries do not care for ethics when using facial recognition tools.... It shows how much ethics are important in anything regarding technology and implementing into human lives.” [Student 31]

VI. DISCUSSION

The increased use of software across all aspects of lives and society has created a need for software engineering ethics. This need is especially acute as new forms of algorithms and AI-based systems become common. Based on the findings from this specific study and also drawing on lessons from using role-play case studies across ten different course offerings [43-44], we now discuss the relevance of the findings.

Identifying Ethical Concerns: In this research study of the use of role-play case studies for teaching software ethics, we found that students could identify different elements of ethical concerns and develop a more nuanced understanding of how software shapes ethics on the ground. Regarding different concerns related to ethics, the cases allowed students to recognize and reflect on the values of transparency, trust, safety, privacy, and so on. The concerns they identified are not exhaustive but are vital to begin thinking about ethical and responsible software engineering.

Situated Aspects of Software Use: In addition to identifying specific concerns, in relation to the pedagogical approach advance here, the students could look at the situation in a contextual manner and understand the complexity inherent in software systems once they are deployed. They further identified and discussed the limits to the use of technology and the need for human mediation when systems are deployed, especially about automation. Overall, students displayed a remarkable understanding of ethical issues related to software and evaluated the course highly in terms of their learning. The cases also gave them a view of the professional world of software engineering, going beyond the knowledge they had acquired in their courses.

Active Learning and Self-Regulation: Consistent with prior work on active learning, we found that when students are asked to engage with the material with the aim to use that knowledge for a discussion, they show a deeper understanding and increased motivation. This was facilitated also by the use of different forms of content – readings, videos, podcasts – and the freedom students were given to do their own research online. The diversity of content allowed students to find ways of learning that worked for them, and as they researched more, the more interesting content they came across. In other words, self-regulation and autonomy, as the literature says, are important factors in student learning, and encouraging that through pedagogical interventions is important. In addition to the active learning component, the focus on role-play provided students both guidance and guard rails in terms of how to engage in a discussion and what to say, but also make them reflect on the different stakeholders involved in the process. This diversified their viewpoints and sometimes led to a change in their perspective.

Novel Aspects of Pedagogical Approach: There was a novelty factor as well both in terms of content and the use of case studies and role-play discussions as students had not yet encountered this form of pedagogy. In relation to the content of the course, the topic and the readings allowed students to engage their whole self, their identity, in the topic and discussion as many of them were aligned in their values with

their roles and with the course topic in terms of creating more ethical and responsible software. In terms of pedagogy, in all their other courses, they were taught in a traditional manner with lectures and if there was group work, it was not directed in any form but was more of a “work as a team” approach. This often resulted in free-riding but in role-play discussion it is difficult to free-ride as everyone has to contribute. Student participation varied, but they all had to participate.

Teaching Insights for Using Role-Play Case Studies:

There are several lessons from this study for using role-play case studies for teaching software engineering ethics: 1) The cases should be designed to be relevant for students, this means either newsworthy topics or those they can related to at a personal level. In our case, Boeing and VW were both newsworthy and facial recognition and lending are relevant to students’ everyday lives. 2) The scenarios should be succinct and leave space for interpretation by students so that they have to struggle with a response and it is relatively open-ended. This makes for a better discussion. In our case, teams often reached recommendations that were in opposition to other teams. 3) Although our findings here are from a single course offering, we realized there is value in creating a simpler form of a case initially and extending it over time with more details if students show an interest in it and there is evidence of learning. 4) Multiple assessment methods are possible for case studies and should be implemented. We used pre/post discussion questions, concept maps, and, in some cases, multiple choice questions, and all are feasible for this approach.

Faculty Effort and Resource Issues: Like most active learning pedagogical strategies, designing and using role-play case studies is a time-consuming intervention. There are several ways though to mitigate the effort required. First, many cases that have been developed by others are available publicly with permission to re-use [43-44]. Second, given the impact this approach has on students and their learning, even a short case and even a single case can contribute to raising ethical awareness among students. Third, once the initial development has been done, the approach is quite flexible and can be easily adapted to other courses and modules if required.

VII. CONCLUSION

In this paper, we present a study of the efficacy of using role-play case studies for teaching software engineering ethics. We found that by participating in this pedagogical intervention students were able to learn about different ethical concerns and about contextual factors affecting ethical use of software. We provide details of student assessment. Case studies and other materials developed as part of this project are available online publicly through open access for others to use.

ACKNOWLEDGMENT

We thank the study participants and the U.S. National Science Foundation Award # 1937950, 2335636 for partial support. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

- [1] C. O'Neil, "How algorithms rule our working lives", *The Guardian*, Sep 2016.
- [2] F. Pasquale, "The Black Box Society: The Secret Algorithms That Control Money and Information", Cambridge, Massachusetts; London, England: Harvard University Press, 2015.
- [3] C. Fiesler, N. Garrett, and N. Beard, "What Do We Teach When We Teach Tech Ethics?: A Syllabi Analysis," in *Proceedings ACM SIGCSE*, Feb. 2020, pp. 289–295.
- [4] N. Garrett, N. Beard, and C. Fiesler, "More Than 'If Time Allows': The Role of Ethics in AI Education," in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, Feb. 2020, pp. 272–278.
- [5] A. Narayanan and S. Vallor, "Why software engineering courses should include ethics coverage". *Commun. ACM* 57, 3, 23–25, 2014.
- [6] G. Génova, M. R. González, and A. Fraga, "Ethical Education in Software Engineering: Responsibility in the Production of Complex Systems," *Sci Eng Ethics*, vol. 13, no. 4, pp. 505–522, Dec. 2007, doi: 10.1007/s11948-007-9017-6.
- [7] J. Hughes et al., "Global and Local Agendas of Computing Ethics Education," in *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, Jun. 2020, pp. 239–245.
- [8] B. B. Bullock, F. L. Nascimento, and S. A. Doore, "Computing Ethics Narratives: Teaching Computing Ethics and the Impact of Predictive Algorithms," in *Proceedings of ACM SIGCSE*, Mar. 2021, pp. 1020–1026.
- [9] J. P. Hertel and B. J. Millis, *Using simulations to promote learning in higher education: an introduction*. Sterling, VA: Stylus Pub., 2002.
- [10] D. Gotterbarn, K. Miller, and S. Rogerson, "Software engineering code of ethics," *Commun. ACM*, vol. 40, no. 11, pp. 110–118, 1997.
- [11] D. D. Gotterbarn and D. K. W. Miller, "Computer ethics in the undergraduate curriculum: Case studies and the joing software engineer's code," *Journal of Computing Sciences in Colleges* 20, no. 2 (2004): 156–167.
- [12] D.D. Gotterbarn, "Software Engineering Ethics," 2001. *Encyclopedia of Software Engineering*, Wiley Online Library." Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/0471028959.sof314>
- [13] ACM, "Code of ethics," 2018, <https://www.acm.org/code-of-ethics>
- [14] IEEE, "Code of ethics," n.d. <https://www.ieee.org/about/corporate/governance/p7-8.html>
- [15] Software Engineering Code of Ethics and Professional Practice: <https://ethics.acm.org/code-of-ethics/software-engineering-code/>
- [16] B.W. Boehm, *Software Engineering Economics*, Prentice Hall, NJ. 1981.
- [17] Y. Lurie and S. Mark, "Professional ethics of software engineers: An ethical framework." *Science and engineering ethics* 22 (2016): 417–434.
- [18] F. B. Aydemir and F. Dalpiaz, "A roadmap for ethics-aware software engineering," in *Proceedings of the International Workshop on Software Fairness*, in FairWare '18. New York, NY, USA: Association for Computing Machinery, May 2018, pp. 15–21.
- [19] R. Alidoosti, P. Lago, and M. Razavian, "Ethics in Software Engineering: a Systematic Literature Review". 2022, Technical Report. <https://research.vu.nl/en/publications/ethics-in-software-engineering-a-systematic-literature-review>
- [20] A. Rashid, K. Moore, C. May-Chahal, and R. Chitchyan, "Managing Emergent Ethical Concerns for Software Engineering in Society," in 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, Florence, Italy: IEEE, May 2015, pp. 523–526.
- [21] J. Singer and N. G. Vinson, "Ethical issues in empirical studies of software engineering," *IEEE Trans. Software Eng.*, vol. 28, no. 12, pp. 1171–1180, Dec. 2002, doi: 10.1109/TSE.2002.1158289.
- [22] J. Greeno, A. Collins and L. Resnick, "Cognition and learning". In R. Calfee & D. Berliner (Eds.), *Handbook of educational psychology* (pp. 15–46). New York: MacMillan, 1996.
- [23] J. Lave and E. Wenger, "Situated Learning: Legitimate Peripheral Participation". Cambridge University Press, 1991.
- [24] A. Johri, B. Olds and K. O'Connor, "Situative frameworks for engineering learning research". In Johri, A. & Olds, B. (ed.). *Cambridge handbook of engineering education research*, 47–66, 2014.
- [25] A. Johri and B. Olds, "Situating engineering learning: Bridging engineering education research and the learning sciences". *Journal of Engineering Education* 100 (1), 151–185, 2011.
- [26] R. K. Sawyer and J. G. Greeno, "Situativity and Learning," *The Cambridge Handbook of Situated Cognition*, pp. 347–367, 2001.
- [27] C. van de Sande and J. G. Greeno, "Achieving Alignment of Perspectival Framings in Problem-Solving Discourse," *Journal of the Learning Sciences*, vol. 21, no. 1, pp. 1–44, 2012.
- [28] D. Martin, G. Bombaerts and A. Johri, "Ethics is a disempowered subject in the engineering curriculum". *Proceedings of SEFI* (2021).
- [29] J. R. Herkert, "Engineering ethics education in the USA: Content, pedagogy and curriculum". *European Journal of Engineering Education*, 25 (4), 303 – 313, (2000).
- [30] J. L. Hess and G. Fore, "A Systematic Literature Review of US Engineering Ethics Interventions". *Science and Engineering Ethics*, 24(2), 551–583 (2018).
- [31] J. R. Herkert, "Ways of thinking about and teaching ethical problem solving: Microethics and macroethics in engineering." *Science and Engineering Ethics*, 11(3), pp.373–385 (2005).
- [32] V. Varma and K. Garg, "Case studies: the potential teaching instruments for software engineering education," *Fifth International Conference on Quality Software (QSIC'05)*, Melbourne, VIC, Australia, 2005, pp. 279–284, doi: 10.1109/QSIC.2005.18.
- [33] M. C. Loui, "Fieldwork and cooperative learning in professional ethics." *Teaching Philosophy*, 23, 139–156, (2000).
- [34] M.C. Loui, "What can students learn in an extended role-play simulation on technology and society?" *Bulletin of Science, Technology & Society*, 29(1): 37–47 (2009).
- [35] R. H. Prince, "Teaching engineering ethics using role-playing in a culturally diverse student group," *Science and Engineering Ethics*, vol. 12, no. 2, pp. 321–326, 2006.
- [36] C. Shaw, "Using role-play scenarios in the IR classroom: An examination of exercises on peacekeeping operations and foreign policy decision making." *International Studies Perspectives*, 5(1), 1–22 (2004).
- [37] D. A. Tamburri, M. Razavian and P. Lago, "Teaching software design with social engagement," 2013 26th International Conference on Software Engineering Education and Training (CSEE&T), San Francisco, CA, USA, 2013, pp. 61–69, doi: 10.1109/CSEET.2013.6595237.
- [38] T. Kubica, R. Peine and I. Braun, "Role-Based Group Formations and Interactions to Foster Collaboration in Different Classrooms," 2020 IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T), Munich, Germany, 2020, pp. 1–2.
- [39] R. G. Epstein, "The use of computer ethics scenarios in software engineering education: The case of the killer robot," in *Software Engineering Education*, J. L. Díaz-Herrera, Ed., in *Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer, 1994, pp. 429–437.
- [40] J. Gogoll, N. Zuber, S. Kacianka, T. Greger, A. Pretschner, and J. Nida-Rümelin, "Ethics in the Software Development Process: from Codes of Conduct to Ethical Deliberation," *Philos. Technol.*, 34, (4):1085–1108, 2021
- [41] C. Marshall and G. B. Rossman, *Designing qualitative research*. 2014. Sage publications.
- [42] A. Strauss, *Qualitative analysis for social scientists*. Cambridge university press, 1987.
- [43] A. Hingle and A. Johri, "Role-Play Case Studies to Teach Computing Ethics: Theoretical Foundations and Practical Guidelines." *Proceedings of HCISS 2024*.
- [44] A. Hingle, H. Rangwala, A. Johri, and A. Monea, "Using role-plays to improve ethical understanding of algorithms among computing students." In 2021 IEEE Frontiers in Education Conference (FIE), pp. 1–7. IEEE, 2021.
- [45] A. Johri, "George Mason Tech Ethics," 2024 <https://onlineethics.org/collection-detail/George%20Mason%20Tech%20Ethics>