Title:
Information Ethics and Social Issues in the Undergraduate Computer Science Curriculum: A Curriculum Development and Implementation Report

Authors:
Melissa J. Dark, Purdue University, CERIAS, phone 765-496-6761, fax 765-496-3181
Ahmad Ghafarian, North Georgia College and State University, phone 706-864-1677
M.A.Saheb, Georgia Southwestern University, phone 229-931-2115
Jesse Yu, College of St. Elizabeth, phone 973-290-4067

Presenters:
Melissa Dark, Ahmad Ghafarian, M.A. Saheb, and Jesse Yu

Author to Contact for Correspondence about the Paper:
Melissa Dark
Purdue University
CERIAS
Recitation Hall
West Lafayette, IN 47907
dark@cerias.purdue.edu

Abstract:

This paper will report on a National Science Foundation funded project designed to advance information security in postsecondary education through the development and integration of information ethics and social issues in the undergraduate computer science curriculum. In this paper, we will report the rationale for the project, methods used to improve the instructional capability of computer science and security faculty with respect to emerging sociological and ethical issues associated with information assurance and security, and methods used to develop and test curricular materials. The authors will share examples of the curriculum materials developed as a result of the project.

Introduction

This paper is the product of a workshop entitled "Protecting Information in the Computer and Beyond," which was organized by the Center for Education and Research in Information

Assurance and Security (CERIAS) at Purdue University and sponsored by the National Science Foundation[1].

The goal of the NSF sponsored project is to improve the instructional capacity of computer science and security faculty with respect to emerging sociological, ethical, and educational issues associated with information assurance and security. In this paper, we present how the workshop has helped to advance information security in postsecondary education through the development and integration of ethics and social issues in the undergraduate computer science curriculum.

Background

Information assurance and security is a ubiquitous and growing concern for many administrative, industrial, academic, political, and economic entities and activities. As the revolution in information reaches every part of our society, we are becoming increasingly more dependent on the information. We are at a time that technical capabilities have advanced more rapidly than consideration of the social effects. That makes addressing the information assurance and security a critical priority. We face the immense challenge of educating the next generation of computer scientists, software engineers, programmers, technicians, and end-users in security practices that are both effective and consonant with the values of the society in which they are to be used. A case in point of the need for this is the ACM Computing Curricula 2001. It calls out social and professional issues as an independent knowledge area and one of the core topics of the computer science body of knowledge (which requires at least sixteen core hours). Unfortunately, many computer science programs do not offer any specific courses or units to address these particular issues. This paper represents the collective thinking of twenty one faculty members from a variety of disciplines (computer science, ethics, communication, linguistics and

[1] This project was sponsored in part by the National Science Foundation Grant Due # 0113725.

educational faculty) regarding methods for integrating ethics, social, and professional issues into the computer science curriculum and methods used to develop and test curricular materials.

Workshop Overview

The goal of the workshop was to increase the knowledge and skills of the faculty, and engage the faculty in curriculum planning, integration, and instructional materials development. Throughout the workshop, the faculty participants worked with faculty mentors who have research and teaching experience in information assurance and security in the disciplines of communication, education, linguistics, and philosophy. The workshop involved multiple disciplines that extend beyond the boundaries of computer science. Faculty participants gained knowledge and skills in the following areas:

- The capacity to do great good or great harm in professional life. Mindfulness of capacities and responsibilities. Topics included: 1) philosophical ethics, 2) the circumstances of action or choice, 3) ethical evaluation, the language of evaluation and moral theory, 4) utilitarianism, 5) neo-Kantian philosophy (the rights view), 5) virtue theory, 6) essentialism, 7) pragmatism, 8) ethics and information, 9) Privacy, Secrecy and Security, 10) Networked Systems, Community and Identity, 11) Ethics and System Design, 12) Whistle blowing and Professional Ethics, and 13) Standard Research Ethics.
- Natural language and meaning, meaning as information, and the implications of language and meaning on information security, applications of ontology, connotation, intention, language use and abuse, deliberate concealment and accidental disclosure, encoding and decoding, and natural language processing on information security. Topics included: 1) extracting and representing meaning of natural language texts, 2) reasoning about knowledge derived from texts, 3) generating texts from meaning representations, 4) basic

problems in NLP, 5) theory, description, and methodology in ontological semantics, 6) kinds of lexical representations, 7) constructing a computational lexicon, 8) automating acquisition, and 9) natural language watermarking.

- Communication as a social act and the role and purpose of information technology in enabling this/these social act(s). Topics included: 1) surveillance trends and practices, 2) the electronic panopticon, 3) the Panoptic effects model, 4) metacommunication, and 5) equity theory.
- Teaching through exposition and inquiry. Regardless of the discipline, students in the field of information assurance need to develop good problem solving skills in order to become successful professionals. This project is incorporating specific learning strategies with the goal of teaching students problem solving skills that will transfer to professional practice. This includes being able to both "problem set" and "problem solve." That is, Information Assurance and Security practitioners must be prepared to analyze complex social, economic, political, and legal situations; to determine which, of many presenting issues, is most critical; to consider multiple consequences to proposed solutions; and to prioritize actions to take. These are skills that are <u>not</u> typically learned in a computer science classroom. To address this, we are developing expository materials that utilize an inquiry approach to learning. Specifically, we are developing several case studies for use in the classroom and in the textbook. Case-based instruction has the potential to prepare students to solve the kinds of complex, ambiguous problems that they will face in practice. In case-based instruction the learning focus shifts from the explicit knowledge and skills that form the traditional academic curriculum to the development of active knowledge. Active knowledge goes beyond simply recalling

information to the ability to use that information to select relevant issues and solve identified problems. According to Ewing [6] of the Harvard Business School, "students change profoundly in their ability to undertake critical analysis and discuss issues intelligently, coming to a greater understanding of the complexity of doing business." Case-based instruction offers a number of advantages for professional education that are relevant to this project. Cases are thought to be more effective than didactic teaching methods because they 1) more accurately represent the complexity and ambiguity of real-life problems, 2) provide a framework for making explicit the problem-solving processes of both novices and experts, and 3) provide a means for helping students develop the kind of problem-solving strategies that practicing professionals use [7].

- Curriculum Development and Integration. Participant faculties as well as other faculties in order to enhance the Computer Science curriculum will use the knowledge and skills gained at the workshop. This process involves creating case studies and using them in their courses as a replacement or supplement of the teaching process for teaching ethical and social issues in undergraduate computer science program.

Curriculum Planning and Integration

Several questions were formulated by the principal investigators to guide the curriculum integration process. The following questions were used to analyze and identify ethical and social issues as well as to analyze methods for integrating these issues into the undergraduate computer science curriculum.

1. What is ethically and socially significant about computers and computing?
2. What are ethical problems/issues in computing?
3. What are social problems/issues in computing?

4. What are the security and information protection issues in computing?
5. What should undergraduate students know about and be able to do in regard to ethical, social, security, and information protection issues in order to be prepared for professional practice?
6. How do we teach these (items 1-5 above) to the next generation of software/system professionals to increase their awareness and knowledge, establish a sense of responsibility, and hopefully, impact (for the better) the quality of the products they produce?

Early in the workshop, the group discussed integration methods. We debated the advantages and disadvantages of teaching these issues in a separate, stand-alone course in comparison to the inclusion of these issues as lessons and/or modules integrated throughout the curriculum. After much discussion (which is beyond the scope of this paper), the consensus was that we would focus on ways to integrate these issues across the curriculum. To address these questions across the curriculum, we aggregated the computing curriculum into three main groups:

1) Introductory Courses (Discrete Structures, Programming Fundamentals, Algorithms and Complexity)
2) Software and Programming (Programming Languages, Graphics and Visual Computing, Intelligent Systems and Software Engineering)
3) Hardware and Systems (Architecture and Organization, Operating Systems, Human Computer Interaction, Information Management, and Net-Centric Computing)

Faculty participants joined one of the three groups based on their teaching expertise and interests. Each group generated a list of recommendations regarding how those topics could be integrated as modules throughout the undergraduate computer science curriculum. The recommendations were then converted to a curriculum framework that is organized by the three main groups, issues within each area, and the learning objectives for each issue [Appendix A].

Participants will have to author a number of instructional case studies and accompanying instructor's guides after the workshop. The case studies are 2-4 page instructional stories designed to help students uncover ethical and social issues in computer science. The instructor's guide will include facilitation questions and debriefing guidelines to assist instructors' effective utilization of the case studies. We want to point out that the curriculum framework [Appendix A] is by no means complete. It is not designed to cover every aspect of the computer science curriculum. It is merely the result of the workshop based on the consensus of all the participants at the workshop.

Instructional Materials Development

One of the main focuses of the CERIAS workshop was the creation of classroom case study materials for integrating ethical and social issues into the computer science curriculum. Case-based instruction is popular in many disciplines including science, art, literature, as well as engineering and is considered to be especially useful for teaching issues-oriented topics. These case studies are to be used by faculty participants as well as by other interested faculties in classroom instruction. In this section we present the methods used to develop the case studies.

Faculty participants are now in the process of writing case studies. The case studies will be reviewed by faculty mentors and peers in the workshop. The case studies will also be pilot tested by faculty participants in their undergraduate courses. Based on the feedback from reviewers and pilot testing in the classroom, the case studies will be revised for future use and broader dissemination.

Project Outcomes

At this stage of the project, we address the project outcome in terms of impact on faculties, case studies, and preliminary results as described below:

1. **Impact on Faculties**. This two-year project is currently in the middle of the first year of operation. To date, seventeen faculty participants have increased their knowledge and skills with regard to ethical and social issues in computing and information security, as well as how to integrate these topics into their undergraduate curricula to impact students. We plan to study the impact of this integration on students as it unfolds and will present these findings when we are further along in the project. The participant faculties are enthusiastic and very interested in the project. This is evident by the amount of progress that has been made so far. Here is a quote from one of the participant faculties.

   "The workshop provided computer science faculty the opportunity to see how the ethical, social, and professional issues could be approached and addressed in other courses such as philosophy, which are not computer science classes. This opens the door for collaborations among various departments to teach these issues in a more broad perspective across multiple disciplines."

2. **Case Studies**. The purpose of the case studies is to pilot test the integration of ethical and social issues into the undergraduate computer science curriculum. Most of the participant faculties showed interest in creation of case studies. Fourteen case studies have been developed so far and there are more under construction. Sample case studies can be found in Appendix B. Here is a quote from one of the participant faculties.

   "I found that an instructional case study is a very effective and useful way to teach students the ethical, social, and professional issues. One or two case studies could be introduced in a computer class to bring the ethical, social, and professional issues to students with each case study taking fairly short period of time to address specific issues.

Through case study, we can teach students to identify problems, raise questions, look for solutions, and justify the reasoning behind their solution and conclusions."

3. **Preliminary Results**. The participant faculties have used some of their own case studies as well as others in their classroom instruction to pilot test this approach of integrating ethical and social issues into the computer science education. The preliminary results have been very encouraging. Both students and faculties have expressed their satisfaction of teaching and learning ethical and social issues. Here is a quote from another faculty who used case studies as an instructional tool for the teaching of ethics and social issues. "I used some studies in my Introduction to Internet, C programming and Database classes to introduce the ACM Code of Ethics and Professional Conduct to my students. Each case study addressed some moral imperatives outlined in the ACM Code of Ethics. The benefit of using case study in my classes was two-folded. First, there is no need to spend a large block of lecture time to cover these issues all at once in one particular course so that it can eliminate the concern of not having time to cover traditional course material. Second, when combined properly with course material, it is more natural to bring up some of those issues to students' attention, even though the textbook may fail to devote any section to the issues. By carefully designing the instructional case studies and introducing them at various stages in the course of the program, we can consistently and systematically teach students to be aware of the ethical, social, and professional issues in the computing profession and let students always be cognizant of their responsibilities. I found this pedagogy actually works well for my classes."

   We would like to point out that we have not measured the outcome of the learning process yet. The project is at the preliminary stage and the reports here represent the pilot testing

of the material. We will present full report of teaching and learning outcome in our future paper.

Conclusion

It is our hope that the initiative from the workshop not only bring the ethical, social and professional issues in computing, which sometimes might be overlooked in some computer science programs, to the attention of the entire computer science society, but also to a larger audience from the related disciplines. In light of the broadening scope of computing, information assurance and security is a collaborative work across multidiscipline. We should start from educating the next generation of computer professionals as well as end-users. The initiative is to help advancing information security in postsecondary education and integrating information ethics and social issues in the undergraduate computer science curriculum. One point worth mentioning is that the curriculum framework [Appendix A] is not inclusive. It is rather a starting point to help faculty to integrate these topics into their curriculum by providing some useful methods as well as good references to be of value to them. It should be visited often and expanded to incorporate any new developments. It is always an ongoing process.

References

[1] L. Aleamoni, "Issues in linking instructional improvement research to faculty development n higher education," *Journal of Personnel Evaluation in Education*, vol. 11, 1, pp. 31-37, 1997.

[2] M. Atallah and V. Raskin. (2000, April). Natural language watermarking: Design, analysis, and a proof-of-concept implementation, Presented at the 4th Information Hiding Workshop. [Online] Available: http://omni.cc.purdue.edu/~vraskin/IHW.AtaRasEtAl.pdf).

[3] C. Botan and M. Vorvore, "What Are You Really Saying To Me?" presented at the International Communication Association Conference, Acapulco, Mexico, 2000.

[4] M. Vorvoreanu and C. Botan, "A Review of Theoretical Perspectives and Research Findings," presented at the International Communication Association Conference, Acapulco, Mexico, 2000.

[5] C. Botan, Communication work and electronic surveillance: A model for predicting panoptic effects," in *Communication Monographs*, 1996.

[6] D. Ewing, *Inside the Harvard Business School*. New York: Random House, 1990.

[7] M. Julian, M. B. Kinzie, and V. A. Larsen, "Compelling case experiences: Performance, practice, and application for emerging instructional designers," *Performance Improvement Quarterly*, vol. 13(3), p.p. 164-201, 2000.

[8] V. Raskin, "Ontology in information security: A useful theoretical foundation and methodological tool," *Pre-Proceedings of the 2001 New Security Paradigms Workshop*, 2001, pp. 49-52.

[9] V. Raskin, M. Atallah, C. McDonough and S. Nirenburg, "Natural language processing for information assurance and security: An overview and implementations," *Proceedings of NSPW-2000*. 2001, pp.51-65.

[10] P. Thompson, “Privacy, secrecy, and security,” *Ethics and Information Technology*, vol. (3), pp. 13-19, March. 2001.

Appendix A

Table 1

Cluster	Issue	Learning Objective
Introductory Courses (IC)	1. Make students aware of the ethical aspects of computing	a. identify the role of ethics in decision making and professional practice
		b. identify the consequences of ethical and unethical behavior
		c. explain the ACM code of ethics
	2. Identify the importance, complexity, consequences, and obligations of computing professionals	a. awareness of competing roles to employers, themselves, society, and their profession
		b. identify stakeholders in an issue and our obligations to them
		c. identify ethical choices
		d. articulate ethical tradeoffs in a technical situation
	3. The impact of computers in society and the wide effects outside computing professions	a. describe positive and negative ways in which computers alter social interactions of people
		b. identify global impacts on diverse cultures
		c. discuss implications of differences in access and utilization to computing resources
	4. Professional obligations in producing work product	a. explain the role of problem set definition and system design
		b. explain the role of honesty in analysis and feasibility
		c. explain the role of identifying risk and liabilities
		d. explain the role of testing in professional practice
		e. explain the role of team and individual responsibilities
		f. identify the concrete actions that professionals can take when faced with ethical dilemmas.
Software and Programming (S/P)	1. The need for accountability	a. take ownership and be able to stand up/speak out, give credit/acknowledgement to other people's work, and protect property
		b. specify strengths and weaknesses in relevant professional codes as expressions of professionalism and guides to decision making
		c. analyze accountability from various perspectives (individual, corporation, society, user, etc)
		d. analyze examples of code that demonstrate the infeasibility of complete testing, give practical measures that can minimize risks/errors in code, develop test plan or test cases
		e. discern the importance of correctness, reliability, and safety in programming
	2. Social context, social responsibility, and the digital divide	a. interpret the social context of a particular implementation
		b. perform analysis of the social impact of software
		c. write a statement role playing various users
		d. force the use of readme files, help files, document codes, and ensure virus free software
		e. identify assumptions and values embedded in a particular design
	3. Future implications of current practice	a. examine historical examples of software risk, their context, origin, intended benefits versus harm, and unintended consequences
		b. discuss implications of software complexity with regard to future trends
	4. Plagiarism	a. distinguish between copyright, patents, and trade secrets.

		b. debate software piracy from the perspective of multiple stakeholders
	5. Teamwork	a. student should be aware of the need to work effectively within a team
	6. Sensitivity to social/cultural/gender issues	a. identify objectionable material.
	7. Risks and liabilities	a. discuss time and safety critical software.
		b. articulate viewpoints on technology, the role of technology to better society within the context of benefits, harm, and competing interests
	8. Appropriateness of student projects	a. define appropriate (for what and for whom).
		b. discuss intellectual freedom and appropriate behavior.
	9. Privacy	a. summarize the legal basis for the right to privacy and freedom of expression in one's own nation.
		b. discuss how privacy concepts vary from country to country.
		c. explain how the Internet may change the historical balance in protecting freedom of expression.
Hardware and Systems	Hardware	Testing a. identify accountability issues in testing hardware, b. identify actions that professionals can take when testing hardware, c. identify ethical choices in hardware testing Design a. identify the need for accountability in hardware design, b. identify the ways that computers alter the social relations of people specifically in the context of hardware design, c. identify actions that professional can take during hardware design, d. identify ethical choices in hardware design, e. identify stakeholders and our obligations with relation to hardware design, f. discuss the implications of hardware design complexity and future trends, g. discuss privacy implications with regard to hardware design. Administration a. identify actions that professionals can take in hardware administration, b. identify ethical choices in hardware administration, c. identify stakeholders in a hardware administration issues and our obligations to them.

	Databases	Testing a. identify accountability issues in testing databases, b. identify actions that professionals can take when testing databases, c. identify ethical choices in database testing. Design a. identify the need for accountability in database design, b. identify the ways that computers alter the social relations of people specifically in the context of database design, c. identify actions that professional can take during database design, identify ethical choices in database design, d. identify stakeholders and our obligations with relation to database design, e. discuss the implications of database design complexity and future trends, f. discuss privacy implications with regard to database design. Administration a. identify the need for accountability in database administration, b. identify ways in which database administration can alter the social relations of people, c. identify actions that professionals can take in database administration, ethical choices in database administration, d. identify stakeholders in a database administration issues and our obligations to them, e. identify implications of administering complex databases with regard to future trends, and discuss privacy implication of administering databases.
	Human Computer Interaction	Testing a. identify accountability issues in testing human computer interaction, b. identify actions that professionals can take with regard to testing human computer interaction, c. identify ethical choices in testing human computer interaction. Design a. identify the need for accountability in the design of human computer interaction, b. identify the ways that computers alter the social relations of people specifically in the context of designing human computer interaction, c. identify actions that professional can take with regard to designing human computer interaction, d. identify ethical choices in designing human computer interaction, e. identify stakeholders and our obligations with regard to designing human computer interaction, f. discuss the design implications of human computer interaction complexity and future trends.

	Networks and Infrastructures	Testing a. identify accountability issues with regard to testing networks and infrastructures, b. identify actions that professionals can take with regard to testing networks and infrastructures, c. identify ethical choices in testing networks and infrastructures. Design a. identify the need for accountability in network/infrastructure design, b. identify the ways that network/infrastructure design can alter the social relations of people, c. identify actions that professional can take during network/infrastructure design, d. identify ethical choices in network/infrastructure design, e. identify stakeholders and our obligations with relation to network/infrastructure design, f. discuss the implications of network/infrastructure design complexity and future trends, g. discuss privacy implications with regard to network/infrastructure design. Administration a. identify the need for accountability in network/ infrastructure administration, b. identify ways in which network/infrastructure administration can alter the social relations of people, c. identify actions that professionals can take in network/infrastructure administration, d. identify ethical choices in network/infrastructure administration, e. identify stakeholders in a network/infrastructure administration issues and our obligations to them, f. identify implications of administering complex databases with regard to future trends.
	Open versus closed systems	Design a. identify actions that professionals can take in designing open versus closed systems, b. identify ethical choices designing open versus closed systems, c. identify stakeholders and our obligations to them when designing open versus closed systems, d. identify the implications of system complexity with regard to future trends when designing open versus closed systems.

	Architecture	Testing a. identify accountability issues with regard to testing architectures, b. identify actions that professionals can take with regard to testing architectures, c. identify ethical choices in testing architectures. Design a. identify actions that professionals can take in designing architectures, b. identify ethical choices designing architectures, c. identify stakeholders and our obligations to them when designing architectures, d. identify the implications of system complexity with regard to future trends when designing architectures.
	Operating systems	Testing a. identify accountability issues in testing operating systems, b. identify actions that professionals can take with regard to testing operating systems, c. identify ethical choices in testing operating systems. Design a. identify the need for accountability in the design of operating systems, b. identify the ways that computers alter the social relations of people specifically in the context of designing operating systems, c. identify actions that professional can take with regard to designing operating systems, d. identify ethical choices in designing operating systems, e. identify stakeholders and our obligations with regard to designing operating systems, f. discuss the design implications of operating system complexity and future trends. Administration a. identify the need for accountability when administering operating systems b. identify actions that professionals can take when administering operating systems c. identify privacy implications that should be considered when administering operating systems.

Appendix B

John Ashley
by
Jesse Yu

Case Study

John Ashley is a software developer for a computer software company in NY. The company designs and develops custom software for a variety of private and public clients. John is in charge of a software development team. For the past three months, his group has been working on a multimillion dollars trading and accounting system for a financial institution. They encountered some very difficult problems in the development process during these three months. Every time it took them longer than expected time to find and solve the problems. That caused the project almost two weeks behind the schedule. During the meeting with the client, John was told that if they could not deliver the software on time, the contract would be withdrawn and their client would leave the company. John's boss Sam also told him a bad news that the company is considering laying off its employees if the company cannot maintain its profitability. Sam promised John that he would help him in any way he could on this project with his limited resources. Sam said if his group could not complete the project and if his client left the company, the entire development team would be laid off and possibly more people from other groups in his department would be let go as well.

After meeting with his client and Sam, he called a group meeting and told his group that they have to work over time without extra pay so that they could finish the project on time. John told his group that they had to do whatever they could to speed up the work and to finish it on time to

avoid the severe consequences. His boss Sam agreed to let one programmer Adam to help his team with the coding. It seemed that everything was moving smoothly.

A few days later, John had another bad news. This time it was not work related, but his wife Susan. Susan is a housewife with their two children Amy of age 9 and Nick of age 6. Susan was just diagnosed with breast cancer in its late stage. Fortunately, his company's medical insurance covers his wife and children. She had the surgery and received proper medical treatment. During those days, except going to the hospital to see his wife and being with his children at nights, John spent all his time with his group on the project. It was the worst time in his life.

The entire team worked so hard, days and nights and many extra weekends. Finally, they finished the project two days ahead of the delivery date, which was a Friday. John decided to ask everybody in the group out including Adam to celebrate on Wednesday night. Everybody was so happy and relieved. They were laughing and talking everything from the difficulties of the project to the many sleepless nights and so on. During the conversation with Adam, John thanked Adam for his help for the project. After a couple of drinks, Adam told John that the reason that he finished his part so easily and quickly was because he worked on a similar project at a competing software company before he came to this company. He used exact software code that was written by his coworkers at that competing company in the project as well as some code he found in a commercially produced software package. John did not say anything.

The next day, John meets with his boss Sam. What should John say about the project?

Case Overview

This case involves a software developer and his personal and working experiences in the process of developing software for a customer. John Ashley is responsible for a software developing team to design and build software for a client. After the software is completed for delivery, he accidentally found some serious problem with the software. John's action will have profound impacts on himself, his wife, his team members and his company. The issues involved are also complicated by many other factors.

Case Objectives

After analyzing and discussing this case, the students will be able to:

- Identify the problems in the case
- Discuss these problems from the perspective of John, John's wife, Adam, Sam, John's software developing team, the client and the company
- Realize the seriousness of the problem
- Realize the consequences to various actions that could be taken
- Recognize the importance of ethics and professional issues
- Study the ACM Code of Ethics and Professional Conduct
- Recognize that ethical and professional conducts are always in contrary to self-interest
- Identify the actions that professionals can take when faced with ethical dilemmas
- Give credit and acknowledgement to other people's work and protect property
- Analyze accountability from various perspectives
- Distinguish between copyright, patents, and trade secrets
- Identify the need to work effectively within a team

Guidelines

The case is designed to bring the real life problems to the classroom so that students can be aware of ethics and professional issues in their profession. Also from the case study, students can realize the importance of software design, development, individual role and impact in a team environment.

- What are the problems?
- How did the problem(s) develop?
- What are the constraints?
- What are possible actions John can take?
- What are the consequences to various actions John could take?
- What is the responsibility of John?
- What is the responsibility of Adam, Sam and other constituents?
- What are the solutions?
- How to avoid similar problems?
- Do role-play to realize John's dilemma

Additional questions

- What could Sam do?
- What could the developing team do?
- What could the client do?
- Should they tell the client what has happened?
- What is the role of Adam in the team project?
- Why did Adam do what he did?

- Analyze the accountabilities from Adam, John, Sam, and the software company
- What are the differences between copyright, patents and open source code?
- What is the problem with software piracy?

References

[1] Association for Computing Machinery. (1997). The First Society in Computing. [Online]. Available: www.acm.org

[2] Communications of the ACM, February, 1993.

[3] M. D. Ermann, M. Williams and M Shauf, *Computers, Ethics and Society*, 2nd ed., M. C. Ermann, Ed. New York: Oxford, 1997.

[4] G. R. Morrison, S. M. Ross and J. E. Kemp, *Designing Effective Instruction* 3rd ed., New York: John Wiley and Sons, 2000.

Steve Blouin
by
Melissa Dark

Case Study

Steve Blouin is a lead software engineer for Database Solutions, a large multinational company that designs and develops custom software for a variety of private and public clients. For the past five years, Steve has been the lead engineer on a software project for Identification Information Systems (IIS). IIS, one of Database Solutions largest clients, is an IT service provider to over 40% of the Motor Vehicle Branches in North America. IIS performs a variety of tasks including articulating software specifications, managing software development projects, overseeing IT installation projects, and systems maintenance.

As the lead software engineer on the IIS project, Steve is responsible for a software development team of 4 other programmers. Up to a year ago, the team included 5 programmers. Last year Lloyd Johnson was laid off as a result of some restructuring at Database Solutions. Lloyd enjoyed working for Steve and over the last four years they had become friends. They hiked together and often got their families together for cookouts, movies, and other recreation. That was up until the time that Lloyd was laid off. Since the layoff, Steve and Lloyd had not seen much of each other. Their once solid friendship had become strained. The last time Steve called Lloyd, Lloyd told him that things were tough and he preferred that Steve not call to find out how his job search was going. Lloyd had been out of work for four months and might have found employment much quicker if his family had been willing to move. However, his mother-in-law's health was not good and the primary responsibility for taking care of her rested with his wife, so his family elected to stay in the area.

The layoff was no fault of Lloyd's. Lloyd was a good programmer and did his job well. However, he was the least senior member on the team. Unfortunately, Database Solutions lost $1.8 million of business they had held for many years. The loss of the contract meant that 30 employees were laid off, of which Lloyd was one. To this day, Steve vividly remembers sitting in the conference room with his boss, Richard Emmons, and Lloyd. Richard is the division manager responsible for engineering. Richard oversaw 6 other development teams besides Steve's with a total a staff of 65. Steve had worked for Richard for several years and had come to know that, for Richard, it was all about the bottom line. Richard is well known for ensuring that every project has a profit margin of 37%. On April 3, Richard emailed Steve to request that he and Lloyd meet with him at 3:30 in the conference room to discuss the IIS Solutions reengineering project. Based on the chain of events with the IIS project, Steve figured that Richard had bad news.

Indeed, Richard did have bad news. IIS had decided to pull 60% of their new contracts from Database Solutions and take them elsewhere due to a glitch in some software that had surfaced in January. On January 12, a Texas citizen named Howard Smith, was pulled over while driving with a burned out tail light. The strange thing about the whole situation was that when the patrol officer went to check Howard's registration in the system, there was no record of Howard Smith having registered his vehicle in Texas and furthermore, there was no evidence that Howard Smith existed. This was the tip of the iceberg. Over the course of the next three weeks, 324 people lost their identities - that is to say, within the Texas system, there was no evidence that these individuals existed.

The Bureau of Motor Vehicles spent several thousand dollars investigating the problem over the next month. In February they called a meeting with IIS executives to let them know that they pinpointed problems in the software that led to the incident. IIS then started its own investigation. The outcome of the investigation was two-fold: Database Solutions lost 60% ($1.8 million) of the IIS business and in order to hang on the remaining 40%, Database Solutions would need to fix the software. In the meeting on April 3, Richard laid off Lloyd, citing the loss of contracts to IIS. Richard also notified Steve that the software error needed to be fixed immediately and permanently. It was clear that there could be no further problems with the software or it would mean the loss of additional business and the termination of other individuals. Richard's parting words to Steve were "This needs to be fixed right and right away if IIS is to continue employing the members of your team, including yourself. There are a lot of people counting on you Steve, including myself."

Unfortunately, Steve knew it would be virtually impossible to guarantee Richard and IIS that the software would never malfunction. Not that it would impossible to reengineer and error proof the code, just that it would be difficult to do with the limited resources that Richard had authorized. Steve knew that asking his team to work day and night with no overtime to come in under budget, but he also knew he wanted to guarantee IIS that they and the citizens of Texas would not be faced with a similar ordeal ever again.

1. Identify the problem(s) in this case?
2. Discuss these problems from the perspective of Steve, Richard, Steve's software team, IIS, and the citizens of Texas.

3. What are the respective rights of Steve, Richard, Howard, and IIS?
4. Who is responsible for the results of a software program?
5. Has Steve acted responsibly? Explain your answer.
6. Has Richard acted responsibly? Explain your answer.
7. Has IIS acted responsibly? Explain your answer.

Case Overview

This case involves three main characters (Steve, a software engineer, Lloyd, a laid off software engineer, and Richard, a division manager) and one minor character (Howard Smith, a Texas citizen whose identity was lost in the bureau of Motor Vehicles IT system). It also involves three organizations (Database Solutions, the software development company, IIS, a national IT service provider that hired Database Solutions to develop software), and the Bureau of Motor Vehicles (a client of IIS and the ultimate client of the software). The case presents a snapshot of a problem. The problem is that a software bug resulted in 325 Texans incurring a loss of identity in the Texas system. As a result of this problem, Database solutions has lost a majority of its business with IIS and is under the gun to fix the error in order to retain the remaining business.

Case Objectives

After analyzing and discussing this case, the students will be able to:

- Identify the problems in the case
- Discuss these problems from the perspective of Steve, Richard, Howard, Database Solutions, IIS, and the Bureau of Motor Vehicles
- Analyze accountability from various perspectives

- Identify potential root causes that contributed to this problem
- Identify potential solutions
- Analyze the impact of software
- Discuss the need for regulations, standards, guidelines, procedures in software engineering

Debriefing Guidelines

The case is designed to bring the real life problems to the classroom so that students can be aware of ethics and professional issues in their profession. This case can be facilitated in a variety of ways. Described here are two methods that I have found most effective to date.

Method #1:

Step 1:

Begin with a group discussion using questions 1-4 at the end of the case. This helps bring to bear the details of the case and the events. When asked to identify the problem in the case, students will tend to cite the obvious. The most frequent response is that the software had an error. The students needs to be encouraged to explain why this is a problem, for whom it was a problem, and whether or not the persons or organizations in the case could have a right to expect reliable software. Questions 2-4 are intended to solicit answers to these more detailed questions.

Step 2:

Ask students to contemplate question #4 individually. Ask them to write their response to questions 4 on a piece of paper. Note, students might identify more than one party that is responsible. Encourage them to list their response in order of most to least responsible. Then

have them share their thoughts with the class. When I have used this technique, students either tend to gravitate toward a hierarchy of the individuals in the organization who should be held responsible or towards the organizations themselves. For example, some students identify various persons (the software engineer, the division manager, the software testing team, etc.) at Database Solutions that should be held accountable, while other students feel will assign responsibility to Database Solutions, IIS, and the BMV for various tasks. This is a great lead in to talking about different types of controls that could, should, or might be considered to assure the quality of software products.

Step 3:

As a follow up to step 2, I split students into small groups (3-6 students per group) and ask them to brainstorm and document the following:

What types of controls should an organization have in place to control the quality of their employees work?

What types of controls should the industry have in place to assure the quality of an organization's product?

Step 4:

Step three usually leads to a discussion of varying levels of control depending upon the type and characteristics of the software and consequences of malfunction. If you want, you can conclude by having the students develop a taxonomy of types of software, level of control needed, and a rationale for their decision. They can do this in narrative or in a chart like the one below:

Table 2

Type of Software	Characteristics/Usage of this Type of Software	Level of Control Needed	Rationale

Method #2:

Step 1:

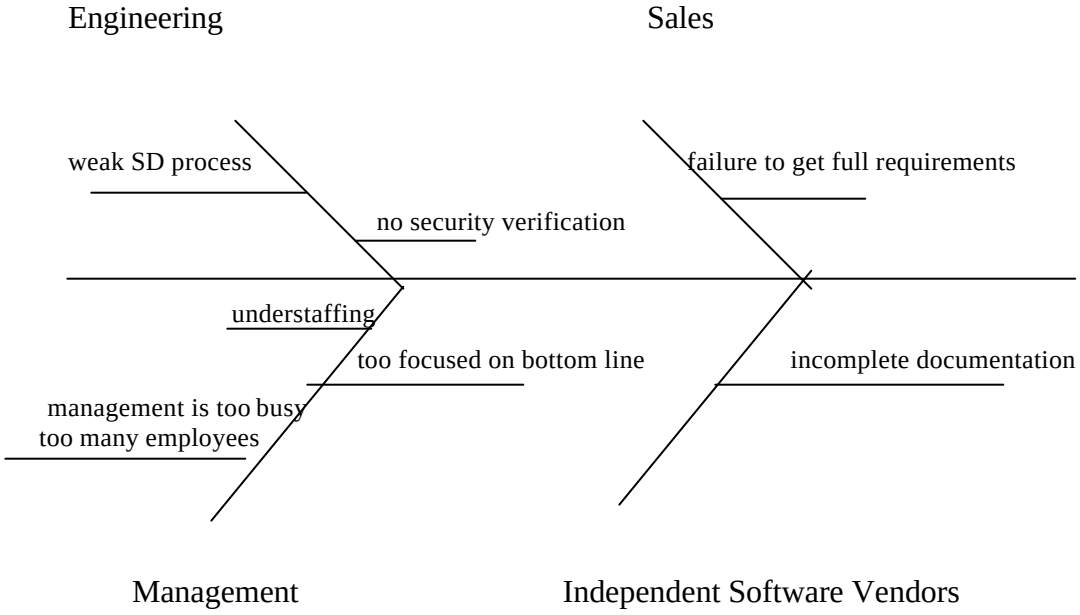
Begin with a group discussion using questions 1-4 at the end of the case. This helps bring to bear the details of the case and the events. When asked to identify the problem in the case, students will tend to cite the obvious. The most frequent response is that the software had an error. The students needs to be encouraged to explain why this is a problem, for whom it was a problem, and whether or not the persons or organizations in the case could have a right to expect reliable software.

Questions 2-4 are intended to solicit answers to these more detailed questions.

Step 2:

In small groups (3-6 students), have students identify all potential contributing factors. Have them document their work on a fishbone diagram. Below is one example of a fishbone diagram that a group of students produced.

Figure 1


Engineering
Sales
weak SD process
no security verification
failure to get full requirements
understaffing
too focused on bottom line
incomplete documentation
management is too busy
too many employees
Management
Independent Software Vendors

Step 3:

Have the groups present their fishbone diagram to the entire class and discuss. This is valuable for students to see each others' work to validate their own thinking and to see what they might have left out.

Step 4:

Again in small groups, have them identify two or three of the biggest problem areas and discuss what countermeasure should be in place to address these problems. Have the students present their conclusion to the class.

Step 5:

Again, this usually leads to discussion of methods of control. Use step 4 from method 1 above or discuss this as a group.

Metropolitan College
by
Ahmad Ghafarian

Case Study

Metropolitan College is a four-year undergraduate college officering bachelor's degree in nursing, science (biology, computer science, chemistry, physics), arts and literature. The school is located in rural areas of state of Nevada. Being a rural private school it has limited budget and therefore they have not computerized all aspects of the college operation yet. One of the centers that perform their work manually is the alumni center. The college produces an alumni newsletter monthly, which needs to be distributed among members. The alumni center director had wanted a computerized alumni system to keep all their records in a database system and generating mailing labels monthly.

Recently one of the local businessmen has donated $50,000 to the college. The college administration has decided to use all or a portion of that money to computerize the alumni center record keeping. The justification was that if the alumni newsletter is published on time and the labels are generated quickly and mailed on time then there is a possibility of generating more money.

Mark Monard who is the director of the alumni center has been given the responsibility of pursuing and monitoring the task. Mark has contacted Information and Instructional Technology (IIT) department to explore the possibility of developing this project. The IIT director after some evaluation has decided to assign Julia Hansen and Kelly Woo to do a system analysis and design and present an overall plan for the project including an estimate of the duration and the cost.

Julia and Kelly immediately contacted Mark the director of the alumni center as well as the registrar and arranged for information gathering through the interview process. Julia and Kelly decided to prepare a set of twenty questions each. At the day of the interview Julia had all her questions typed and ready to take part in the interview. Kelly had all her questions typed and saved in her disk had plan to print them at the interview day. Just half an hour before the interview Kelly tried to print her file. Unfortunately the system could not open the file. Her file must have been corrupted. Because there was no time to create a new set of questions, they had left with two options. Their first option was to postpone the interview to some other time and their second option was to go ahead with interview without Kelly's questions. Since postponing interview may leave a bad impression they decided to go ahead to the interview meeting without Kelly's questions.

After conducting the interview, Julia and Kelly went ahead with systems analysis and design phase and prepared a recommendation for a new alumni system with on-line update and query features. They discussed their ideas with their boss and several other colleagues and they all felt that their ideas were excellent. Julia and Kelly then prepared their system requirements specification nicely and sent one copy to the registrar and another copy to Mark Monard the director of alumni center for their review.

Mark discussed the project with the registrar and asked his opinion about the details of systems. Mark and the registrar agreed to invite the president, all the administrative vice presidents, and the deans of all schools to the presentation. Mark wanted to be sure that he would have the support of the administration and funding for the new on-line alumni system. All

the clerks who were involved with the alumni system and record keeping were also invited to the presentation.

Unfortunately Julia's car ran out of gas the morning of the presentation on her way to college. Consequently she arrived half an hour late for the presentation and noticed that some of the audiences have already left. She immediately apologized and began to set up their presentation with Kelly. They had all the transparency masters ready turn on the overhead projector. They started the presentation by putting their slide on the overhead and describing their system to the audience.

Unfortunately the overhead projector's bulb went out during the presentation and Kelly rushed out to the secretary's office to seek for another bulb. When she finally returned with a new bulb she noticed that half of the audiences have already left. They did finish their presentation but with little of their original enthusiasm.

Case Background

This course involves the development of a computerized alumni system for a college. This is in response to the fact that creating mailing labels manually and mailing the alumni newsletter to the members is done with a delay and thus it is not as effective as the administration of the college would want to be. The main focus of the system is on teamwork to analyze the systems requirement. Julia and Kelly the two employees of the college's IIT department are in charge of the project and must prepare their system requirements to gain administrative support

Case Overview

Metropolitan College is a small liberal arts school that is located in the rural areas of state of Nevada. The college administration including the registrar, alumni center director decided to propose the creation of a computerized alumni center's operation. Two of the staff of the IIT department Julia Hansen and Kelly Woo have been assigned to perform this task. In addition to software engineering knowledge and problem solving ability they should also apply their knowledge of the teamwork and organizational planning. They have made significant effort to prepare their system requirements but they lack on good teamwork to present their presentations to their audience.

Case Objectives

This case is appropriate for use in introductory computer science courses. Frequently, computer science students fail to learn the importance of teamwork in their undergraduate studies. However, effective team functioning is a skill that most employers value. Therefore, after analyzing this case, students will be able to:

1. Identify the impact of individual actions on the outcome of team projects
2. Identify attributes of teams that function effectively
3. Recognize the importance of effective coordination and communication among team members
4. Identify all the resources that are needed for effective presentation and make sure that they are all working

5. Recognize that presentation a project to audience is actually selling yourself and your credibility
6. Recognize the importance of having an alternative plan for your project presentation
7. Recognize the importance of having a practice of your project presentation in advance

Debriefing

This case is designed to encourage students to consider (1) the importance of teamwork in system development (2) the importance of team planning in system development (3) what the issues are when a team of systems analyst plan for presentation of their project (4) realize the rule and importance of software and hardware equipment that are needed during presentation.

Relevant Data

The actual data for this project are registrar, director of the alumni center, IIT department, Julia and Kelly, systems analysis, alumni center, the alumni center computerized system.

Questions

1. Identify the problems in this case?
2. Discuss the problems from the perspective of Julia, Kelly, Mark and the registrar
3. What Julia and Kelly should have done to avoid the problems that have occurred in this scenario?
4. Has Julia acted responsibly? Explain your answer
5. Has Kelly acted responsibly? Explain your answer
6. What do you think Julia and Kelly should do now to try to salvage the alumni project?

7. Do you think they should be given the alumni project?

References

[1] G. Shelly, T. Cashman and H. Rosenblatt, *Systems Analysis & Design*, 4th ed. Boston Massachusetts: Course Technology, 2000.

[2] N. Wetherbe and J. Vitalari, *Systems Analysis and Design Best Practices*, 4th ed. Hightstown, New Jersey: 1995.

[3] P. Kendall, *Introduction to Systems Analysis and Design: A Structured Approach*, 2nd ed. Columbus, Ohio: WCB Pub. Co., 1998.

[4] J. Kendall and J. Kendall, *Systems Analysis and Design*, 3rd ed. New Jersey: Prentice Hall, 1997.

[5] J. Martin, *Systems Analysis and Design of Business Information Systems*, 2nd ed. New Jersey: Prentice Hall, 1995.

[6] J. Hoffer, J. George and J. Valcich, *Modern Systems Analysis and Design*, 2nd ed. New Jersey: Prentice Hall, 1999.