Societal-Consciousness in the Computing Curricula: A Time for Serious Introspection

S. Ramaswamy*, R. Seker*, S. Sudarsan*, D. Moix+

Knowledge Enterprises for Scalable Resilient Infrastructures
Department of Computer Science, University of Arkansas at Little Rock, Little Rock AR 72204
{srini, rxseker, sdsudarsan}@ualr.edu
+*Computer Science Department, Arkansas School for Mathematics, Science and Arts*
moixd@asmsa.org

Abstract

This paper addresses the growing need for inculcating appropriate ethics within the computing curriculum by fostering the development of a societally-conscious ethical framework among our students to address the use of information technology vis-à-vis government, business and society. We propose a new integrated model based approach (IDEA) and suggest its adoption to encourage students on reflecting upon the social and ethical ramifications of technology, beyond the narrow, project-focused tunnel vision that currently (subliminally) exists in many computing curricula, and in particular, in today's profit-focused, consulting and contract-based software industry.

1. Introduction

Computer Science and Engineering education faces significant challenges, such as rapid technological advances, changes in student demographics, and problems associated with the higher education funding. Currently there is an increase in the total worldwide investment in research and a wider worldwide distribution of research activities. The digitalization of work and the reorganization of work processes across many organizations have resulted in routine and/or commodity components being outsourced. It is predicted that government action and economic factors will result in more global competition in both lower-end software skills and higher-end endeavors such as research. It has been argued that standardized jobs are more easily moved from developed to developing countries than are higher-skill jobs. Good educational preparation is one of the primary means for us to prepare our workforce to compete globally for highly skilled technology based jobs. Hiring workers who have been trained to understand international issues will provide our business organizations the necessary competitive advantage in a global market. As the Internet continually eliminates geographic boundaries, the concept of doing business within a single country is giving way to companies focusing on competing in an international marketplace. In this context, we in the teaching profession have the most unique opportunities to become true global visionaries and be highly effective in providing fundamental opportunities for our students that all of our employers so desperately crave and desire. For example, conducting business in another country can be riddled with cultural issues that require deft personal touch, for example: demonstrating appropriate hospitality.

This widespread globalization of the job market calls for our educational institutions to be adaptive, to be robust, and nurture future students by teaching them to work effectively. Not only do these students need to understand and deal with the increased knowledge expectations in the workforce, but must also understand and deal with the increasingly dominant role of technology within their chosen fields and operate effectively in an increasingly multi-cultural and multi-ethnic, global environment. In such jobs, softer skills, which relate to "how we go about getting things done," being language, society and culture-sensitive, etc., are becoming equally important as the hard functional skills (e.g., programming, problem solving, etc) that have traditionally defined what it means to be competent in a chosen field. Educating our future science and engineering workforce to understand such issues in a global context is therefore becoming a highly sought-after experience and a critical differentiator in the employability of our students, often testing their ability to bridge discipline-specific theoretical research issues with real-world practice. The rapidly emerging global economy is profoundly affecting the employment patterns and the professional lives of our graduates. While it has been widely reported that despite intensifying competition, offshoring between developed and developing countries can benefit both parties, many American students have shunned computer sciences and engineering because they fear that job opportunities and salaries in these fields will decline. We need to provide students with higher-order technological skills aptly blended with the consideration of emerging social needs across the globe to provide much needed experiences to thrive in the future as well as be frontline contributors to the technological workforce.

Thus, the future of the computing profession is not in merely our collective ability to graduate good application-level programmers or software designers – these skills have now become commodities that can be outsourced. It is more a reflection on our ability to graduate students who are highly comfortable with the theory but can appropriately blend it with necessary practice by understanding both the business culture and the social issues involved; while being able to effectively share, communicate, articulate and advance their ideas for an innovative product and/or solution. Hence how we prepare our students to be successful employees and entrepreneurs in the global economy is an important consideration, often better taught through the use of appropriate case studies. Students need the exposure to develop a broad, yet pragmatic vision of the changing technological landscape, and need adequate opportunities to develop the necessary soft-skills for being successful in the global workplace. Such an exposure in an academic environment will vastly benefit students who may very well be charged in the near future with developing policies, priorities, and making investments that can help the country remain competitive in the global software systems and services industry.

Hence, a fundamental change in the education of future computer scientists is necessary to insure that they are well prepared for their more professionally demanding roles. These demands relate to success in the job market, responsibilities toward employers, customers and society, and responsibilities as developers of powerful and pervasive technologies. In addition to strong technical and management skills, future software designers need the skills to design customized products and integrated services that meet the diverse needs of a multi-cultural, multi-ethnic, and increasingly smaller world united by rapid technological advances. These advances come with unforeseen challenges and tremendous opportunities. We believe effectively integrating social-responsibility into the computing curriculum is becoming highly critical for the employers of our students.

Many computing curricula, in response to this growing industry need, have placed emphasis on team-based projects and problem-based instruction styles. Often students in such classes epitomize software development as building the best solution to address customers' needs. Through this paper we promote a dilemma-based case study approach that goes beyond a project-based curriculum; thereby encouraging students to reflect upon the social and ethical ramifications of technology, beyond the narrow, project-focused tunnel vision that currently (subliminally) exists across many computing curricula, and the software industry, in particular. Hence, this is an attempt to address some specific concerns that arise out of such a problem/project focused computing curricula. With respect to software related issues, some of these concerns include: *(i)* In today's post-scandal business climate, additional scrutiny, public condemnation, and possible legal consequences could result if individuals and companies continue to violate accepted ethics and fairness standards. While it is often difficult, if not impossible to predict the future, or the negative consequences of a creation, is ignoring such possible consequences ethically questionable? *(ii)* Is it responsible software development practice to create new technology ethically sound enough to disregard any possible negative consequences of the new creation and its effects on society?

The paper is organized as follows: Section 2 discusses dilemma based learning and a partial list of various online resources with specific case studies for incorporating computing ethics in to the undergraduate curriculum. Section 3 proposes the IDEA approach to integrating ethics into appropriate core courses within the curriculum and outlines some examples of how it can be fostered across the curriculum. Section 4 concludes the paper.

2. Dilemma-based Learning

Case-based learning has long been used in business schools [1-4] around the country. It has also proved to be highly effective in other disciplines [5-17]. According to Ewing, "Students change profoundly in their ability to undertake critical analysis and discuss issues intelligently"[18]. Case-based instruction offers a number of advantages and is effective for increasing student motivation [7-8]. In summary, it is thought to be more effective than didactic teaching methods because real-world cases *(i)* more accurately represent the complexity and ambiguity of problems, *(ii)* provide a framework for making explicit the problem-solving processes of both novices and experts, and *(iii)* provide a means for helping students develop the kind of problem-solving strategies that practicing professionals need [19]. Problem-based learning, a case-based derivative, is also widely used, where students are required to learn and apply assimilated knowledge [20]. It is reported to broaden students' views and cause a new awareness of their own ideologies and capabilities and affects growth, questioning, or affirmation [14].

In dilemma-based learning [21-22], another case-based derivative, a story or game is used to communicate the feeling of real-life dilemmas, while challenging its users to learn from the results of their actions. Dilemmas are chosen for their relevancy to complex and costly situations that are difficult for people to comprehend. For example, dilemmas may reflect the complexities of network implementations or the impact of blame on team productivity and project costs. Dilemmas in the classroom challenge learners to balance trade-offs between short-term rewards and long-term results [22]. In prior works, it has been noticed that discussions on

real-world topics [23] through dilemma-based case studies that couple logical investigative thinking of the problem-based approaches with strategic needs assessments – cost, performance metrics, etc. – make appropriate sense in motivating CS students [23-25].

Use of enthusiasm, empathy, and role-play by students has also been shown to be beneficial in improving overall student attitude and encouraging more participation by women students and minorities [26-27]. It helps them develop learning communities and other forms of peer support structures, while emphasizing the positive social benefits of computing. It instills a good feeling among students and motivates them to be participative [28-29]. Hence, a secondary effect of this proposed approach is to help student retention efforts in our programs, as they explore related technology issues and interests in the various domains based upon their own personal analogical contexts and experiences. Thus, a recurring dilemma-based approach integrated into multiple CS classes, could help increase retention of acceptable ethical standards among students regarding computing technology and help them better understand different ethical issues and perspectives.

Dilemma-based instruction, by adopting and building upon themes that dominate our everyday lives, can not only have the greatest impact on subsequent classes, but also help correct the bad blame-driven rapport that the computing discipline has received since the 2001 market crash. Progressive refinement of knowledge gained through more dilemma-based cases in different classes throughout the curriculum will provide the natural progression necessary for the retention of ethical issues, while allowing for reinforcement learning through similar dilemmas, but with increasing technical content of cases. Currently, for interested educators, there are several archival case resources on ethics with appropriate real-world cases that can be easily adapted to the needs of a particular class [30-38]. While none of these provide adequate guidelines for their adoption, use and assessment in CS classes; they can be excellent resources to start the building dilemma-based case studies across several of our core Computer Science classes.

3. The IDEA Model-based Approach

There are two primary players and four steps to the IDEA model. The players include the teachers involved in teaching, and the students involved with learning from, the courses. The four steps are in turn specific to these players. The model is presented in Figure 1. These four steps are explained in the rest of this section with appropriate examples.

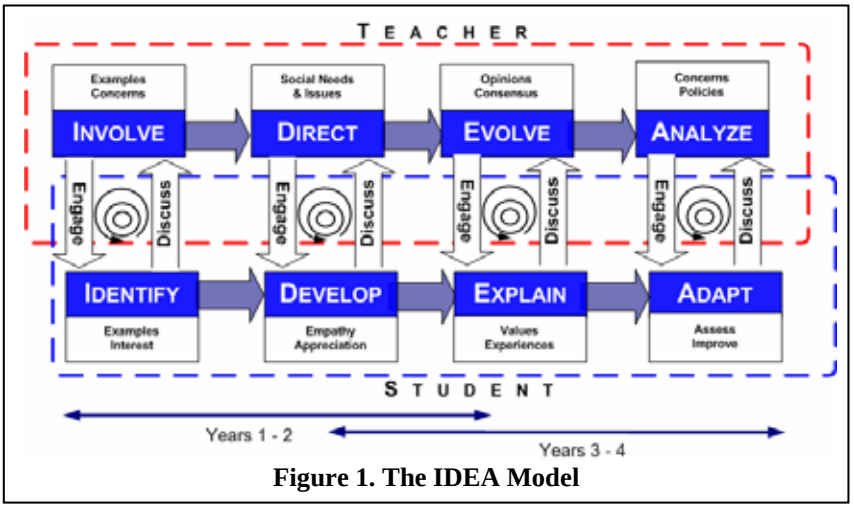

Figure 1. The IDEA Model

IDEA Step 1: Involve & Identify

As illustrated, from the teacher's perspective, the 'I' in IDEA stands for Involve and from the students' perspective it stands for Identify. The teacher begins by involving the student by engaging in a discussion of specific cases that are related to the topic being discussed in the class. For example, in an introductory programming course the discussion may be based on a case that is related to the issue of outsourcing. The teacher presents various concerns with respect to the case in question while at the same time engaging the student's interest through discussions (several societal issues can be discussed here: these include job loss, immigration issues, changing business culture, companies relocating to other countries, etc.). By engaging the students in the identification of appropriately interesting cases (the 'I' for identify in the IDEA model (from the students' perspective)), the students become active participants in the class discussions and hence are more likely to engage in investigating the case study further from various socially-interesting perspectives.

A case study on outsourcing provides the ideal opportunity to dispel some of the pervasive myths that seem to sway students in their choice of computing as a career. Hence for the rest of this paper we will use this as an engaging example to illustrate our model. However, this example is by no means meant to be restrictive; other such examples may be issues of poor GUI design, issues with electronic voting machines, issues of multi-language support in browsers, etc.

IDEA Step 2: Direct and Develop

In step 2, most possibly in a follow up class, the student is directed by the teacher to explore some specific issues of the case further to develop a deeper understanding of

the various issues involved. Going by the outsourcing case study we identified earlier, say an assembly language programming class, students can be engaged in discussion of software outsourcing for embedded systems, say the development of software modules such as drivers that are further integrated into our everyday systems. Issues of security and privacy that are affected by these low-level software modules can be discussed. This allows the student to develop a mental model of the entire issue as well as understand some of the trickier issues in the globalized system of software development, as well as appreciate the finer details of even studying a subject such as assembly language programming and its need in a computing curriculum. Often times, it has been the experience of the authors that students tend to develop a 'follow the herd' mentality and are swayed by what they see and hear as requisite job skills. We have seen that students often espouse the clouded view that they need to spend most of their time in the program learning 'marketable' skills - such as the next hot programming language or system. Hence they should not spend time learning issues that may not be directly related to their future jobs. Hence although highly relevant to learning the fundamentals of Computer Science, courses such as Assembly language programming, evoke less interest among these students. Hence integrating such a case study driven discussion can help assure the students of the need for focusing on such fundamental courses as well as understand its high relevance to societal needs – for example helping build privacy and security in I/O drivers and embedded systems.

IDEA Step 3: Evolve and Explain

In step 3, the 'E' in IDEA stands for 'Evolve' from the teacher's perspective, and 'Explain' from the students' perspective. The student, in the same (Assembly) or a follow up class (say a Database Systems class that normally appears in a junior / senior year of the curriculum) is directed by the teacher to explore more details of the case to understand the magnitude of the various issues involved. Again, taking the issue of outsourcing, the teacher can engage the students in cases such as 'credit card sales and marketing'; whereby the jobs of identifying and seeking 'likely' credit card customers are outsourced to BPO companies (Business Process Outsourcing). Foreign governments are offering significant fiscal and non-fiscal incentives to attract such foreign direct investments into their respective countries and hence it is difficult for a business to ignore such compelling benefits. Experts who see the growing global demand for BPO (estimated to be at $180 billion in 2010) indicate a shift from cost-effectiveness to issues of skills, quality and competence. Issues of personal, professional and business ethics would be factored as we move toward meeting such expectations, often driven by concerned citizens whose personal data is often at stake as part of such BPO decision processes in multi-national organizations.

In a course such as database systems, the teacher can guide discussion on how such practices affect the compilation, sharing and administration of the data contained in large scale databases in question; their effect on issues of an individual's privacy, which possibly is no longer within the geographical confines of the source country and issues of checks and bounds verifications that need to occur for such business arrangements between business operating in two different countries that are culturally different. How is an individual's right to privacy different across cultures and what does 'privacy' mean in a different society? What are the issues a business needs to be concerned with with respect to the laws of the country? How can the business contain and secure the assimilation and sharing of such data? Teachers can promote student discussions that can actually engage the student in understanding core values that may be viewed differently across cultures and growth by discussing cases that involve such experiences.

IDEA Step 4: Analyze and Adapt

As a result of earlier course work, students will have developed the maturity needed to understand the various issues in computing and their relationships, as well as the socio-cultural effects of the numerous aspects of the field. In an appropriate junior/senior level course, students can focus on better understanding the design and development issues requiring global standardization. Such issues include potential misuse of data and the need for software that reduces that potential. Students will be prepared to understand and discuss professional code of ethics issues, since they will have developed a deeper understanding of the need for such a code through their study of good/bad policies and ideas and global business practices.

4. IDEA in Shorter Programs

The IDEA model presented here, though designed for an undergraduate program that spans eight semesters, was successfully adopted in a high school setting by compressing and accelerating the steps. Similarly, this model could be successful in other two-year programs.

Gifted students aged 16-18 in a four-semester residential high school computer science program in Arkansas responded well to the model. It was important to have the students begin thinking about social implications early on. In the first week at the school, students were asked to consider the impact that their use or misuse of the school's computing resources (bandwidth, hardware, network storage) might have on their peers during an

orientation program. In introductory courses in computer programming, networking, database, and web application development, students are consistently confronted with social issues. In a senior-level course in ethics and technology, students are given a more formal, focused environment in which to study and explore these issues as well as explain their views.

One hurdle that is easier to overcome in a smaller, shorter program is buy-in from all faculty members. Communication is easier in a smaller program and the bonds between faculty and students are often tighter when the student-teacher ratio is smaller.

5. Conclusions

In this paper we have presented the IDEA approach to integrating ethics and socially responsible computing into the undergraduate (and high school) Computer Science curriculum. Obviously, for the successful adoption of this approach, one needs the participation of several faculty members from across the department. However, the attractiveness of this approach is in its ability to engage the students meaningfully while still undertaking the primary job of teaching the students the skills they would need to be successful computing professionals upon graduation. For example, faculty subgroups may, during their periods of assessment and preparatory meeting, follow a 'thematic' approach to this process, identifying a specific theme or a related set of themes that is most appropriate in time. They could identify emerging societal issues that would be attractive from a student's personal perspective – for example issues of software development, accountability and related database issues that arise in electronic voting. This is a flexible approach that can be effective in small and large programs at both the high school and university levels.

6. References

1. _____, "*Taking Sides: Clashing Views on Controversial Issues in Business and Society*" edited by Lisa H. Newton and Maureen M. Ford. The Duskin Publishing Group, Inc., Guilford, CT, 1996.
2. Tom L. Beauchamp, "*Case Studies in Business, Society and Ethics*", Prentice-Hall, Englewood Cliffs, NJ 1998.
3. _____, "*Case Studies in Business Ethics*", Edited by T. Donaldson and A. R. Gini, Fourth Edition, Prentice Hall, New York, 1995.
4. R. Spinello, "*Case Studies in Information Technology Ethics*", Second Edition, Prentice Hall, New York, 2003.
5. G. Easton, "*Learning from Case Studies"* Prentice Hall, London, 1983.
6. L. L. Lynn, "*Teaching &Learning with Cases"*, Chatham House Publishers, New York, 1999.
7. B. B. Levin, "Using the Case Method in Teacher Education: The Role of Discussion and Experience in Teachers' Thinking About Cases*", Journal of Teaching and Teacher Education*, V. 10, No. 2, 1996.
8. J. Kleinfeld, "Changes in Problem Solving Abilities of Students Taught Through Case Methods" *1991 Annual Meeting of the American Educational Research Association*, Chicago, 1991.
9. J. L. Kolodner, "Educational Implications of Analogy: A View from Case-Based Reasoning*", American Psychologist*, V 52, No. 1 57-66, 1997.
10. R. Silverman, W. M. Welty, "Teaching with Cases", *Journal on Excellence in College Teaching*, 1990
11. S. Wassermann, "Getting Down to Cases: Learning to Teach with Case Studies", *Teachers College Press*, New York, 1993.
12. C. R. Christensen, A. J. Hansen, L. B. Barnes, "*Teaching and the Case Method"*, Harvard Business School Publishing Division, Boston, Third Edition, 1994.
13. S.M. Williams, "Putting case-based instruction into context: Examples from legal and medical education", The *Journal of the Learning Science*, Vol. 2, pp. 367-427, 1992
14. J. Cossom, "Teaching from Cases: Education for Critical Thinking" *Journal of Teaching in Social Work*, Vol. 5, No. 1, pp. 139-155, 1991.
15. M. Albanese, S. Mitchell, "Problem-Based Learning: A Review of Literature, Outcomes and Implementation Issues*"*, Academic *Medicine*, 68, 52-81, 1993.
16. G. R. Norman, H.G. Schmidt, "The Psychological Basis of Problem-Based Learning: A Review of the Evidence", Academic Medicine, V. 67, No. 9, pp. 557-565 1992
17. H.G. Schmidt, "Foundations of Problem-Based Learning: Some Explanatory Notes" Medical Education, V. 27, pp. 422-432, 1993.
18. D. W. Ewing, "Inside the Harvard Business School", Random House Publishers, New York, 1990.
19. M. J. Julian, M. B. Kinzie, V. A Larsen,., "Compelling Case Experiences: Performance, Practice, and Application for Emerging Instructional Designers", *Performance Improvement Quarterly*, Vol. 13, No. 3, 164-201. 2000
20. L. Wilkerson, W. H. Gijselaers, "Bringing Problem-Based Learning to Higher Education: Theory and Practice: New Directions for Teaching and Learning", December 1996, Jossey-Bass Publications (a John Wiley Company).
21. M. Dark, "*Learning and Teaching with Case Studies*", Center for Education and Research in Information Assurance and Security, Purdue University, IN. http://www.cerias.purdue.edu/education/post_secondary_education/undergrad_and_grad/curriculum_development/ethical_social_prof_issues/learning_and_teaching.php
22. _____, "*Dilemma-Based Learning*",

http://www.brainshift.com/brainshift_01.pdf (Last Retrieved: March 2007)

23. T. Brignall, S. Ramaswamy, "Impact of Different Teaching Paradigms on Student Learning in Technically Intensive Courses: Observations from a Software Analysis and Design Course" *Submitted to International Conference on Information Technology: Coding and Computing (ITCC 2003), Track on Information Technology: Education and Curriculum*, Las Vegas, Nevada, April 28-30, 2003.
24. _____, *"A Framework for Integrating Ethical and Values-Based Instruction into the ACM Computing Curricula 2001"*, Workshop entitled "Protecting Information in the Computer and Beyond." CERIAS, Purdue, Sponsored by NSF, Oct 2001.
25. M. J. Dark, A. Ghafarian, M. A. Saheb, and J. Yu, "Information Ethics and Social Issues in the Undergraduate Computer Science Curriculum: A Curriculum Development and Implementation Report", *6th National Colloquium for Information Systems Security Education (NCISSE)*, Redmond, Washington, June 4-6, 2002. At: http://www.cerias.purdue.edu/education/post_secondary_education/undergrad_and_grad/curriculum_development/ethical_social_prof_issues/framework.php
26. P. James, "Transformative Learning: Promoting Change Across Cultural Worlds", *Journal of Vocational Education and Training*, Vol. 49, No. 2, pp. 197-219, 1997.
27. Lave, J. "The Culture of Acquisition and the Practice of Understanding", in *Cultural Psychology: Essays on Comparative Human Development, Edited by* J.W. Stigler, R.A. Schweder & G. Herdt, Cambridge Univ. Press, pp. 309-327, 1990.
28. J. M. Cohoon, "Recruiting and Retaining Women in Undergraduate Computing Majors", Inroads SIGCSE Bulletin, V. 34, No. 2, pp. 48-52, 2002.
29. K. Treu, A. Skinner, "Ten Suggestions for a Gender-Equitable CS Classroom", Inroads SIGCSE Bulletin, V. 34, No. 2, pp. 165-167, 2002 (Previously appeared in The Journal of Computing in Small Colleges, V. 12, No. 2, pp. 244-248, 1996)
30. http://catless.ncl.ac.uk/risks (Last Retrieved: March 2007)
31. http://onlineethics.org/ (Last Retrieved: March 2007)
32. http://privacyrights.org (Last Retrieved: March 2007)
33. http://computingcases.org/index.html (Last Retrieved: March 2007)
34. http://seeri.etsu.edu/Ethics.htm (Last Retrieved: March 2007)
35. http://ethics.iit.edu/resources/onlineresources.html (Last Retrieved: March 2007)
36. http://www.cyberlawclinic.org/casestudy.htm (Last Retrieved: March 2007)
37. http://engineeringethicsblog.blogspot.com/ (Last Retrieved: March 2007)
38. http://csethics.uis.edu/dolce/ (Last Retrieved: March 2007)