CS2023: Computing for Social Good in Education

Heidi J.C. Ellis, Gregory W. Hislop, Mikey Goldweber, Samuel Rebelsky, Janice Pearce, Patti Ordonez, Marcelo Pias, Neil Gordon

Computing for Social Good in Education (CSG-Ed) engages students with the positive potential of computing to benefit society. It can introduce students to aspects of professional responsibility, which computing students need more than ever given the extensive impact of computing on organizations and individuals. CSG-Ed also has many connections to the CS2023 Computing Curriculum emphasis on Society, Ethics, and the Profession. This article introduces CSG-Ed, discusses the development of CSG-Ed, and provides examples and best practices for CSG-Ed in undergraduate computing programs.

Introduction

Computing continues to expand world-wide and computers have become essential in modern society. This rapid development has provided many examples showing that the intent and impact of computing can be either helpful or harmful. For the purposes of this discussion, it is useful to consider several concepts in describing computing. The first is **Computing for Social Good** (CSG). CSG is the application of information and computing technology for societal development in areas such as healthcare, education, poverty reduction, and sustainability. The United Nations Sustainable Development Goals and similar frameworks provide good examples of the scope of CSG. CSG is computing where the primary intent is to improve people's lives and address societal problems.

The second concept is **harmful computing**, where the intent of the computing activity causes harm. Examples include cybercrime, invasions of privacy, and propagation of misinformation [6].

A third relevant concept is **socially responsible computing**, which addresses avoiding harm in computing. Socially responsible computing includes topics such as unbiased algorithms, explainable AI, computing security, privacy, reliability, and equitable access to computing. The goal for socially responsible computing is to "do no harm", while the goal for CSG is to "do good".

Computing educators should provide computing students with an appreciation of the range of computing impacts including the potential of computing for social good. Students need this understanding as part of becoming socially responsible professionals. CSG is also valuable in education since there is evidence that the positive impacts of computing may help to attract more women and students from other underrepresented groups to study computing.

The remaining sections of this paper provide more detail on how CSG has developed over time, several categorizations to frame use of CSG in education, a series of examples of CSG in undergraduate computing programs, and a summary of best practices for CSG in education. In addition to extending prior work, this paper considers CSG-Ed in relation to the CS2023 Curriculum Recommendations, and provides a discussion of barriers to adoption of CSG-Ed.

What is Computing for Social Good in Education?

Computing for Social Good in Education (CSG-Ed) incorporates concepts and elements of CSG into computing education. CSG-Ed can motivate and engage students by introducing them to authentic examples of the positive potential of computing. CSG-Ed can also provide a context to facilitate development of students' sense of professional responsibility and to address the ethical and social impacts of computing.

Why should we teach CSG?

CSG-Ed is not a new idea. Ben Shneiderman highlighted the social relevance of computing in 1971 [69]. He called for educating students about how computing can be utilized for the common good. However, since that time, educators have never fully embraced CSG-Ed in spite of a series of group efforts and significant publications [36, 39]. In this section we describe a number of reasons why CSG-Ed should be a significant focus in computing education.

Social responsibility is a broadly recognized concept across professions. In medicine, various formulations of the Hippocratic Oath represent the obligation in medicine to create social good and avoid harm. [4] Professional practice in law has a well-established concept of unpaid work, "pro bono publico", "for the public good", and professional societies for lawyers set explicit targets for pro bono work. For example, the American Bar Association model rule 6.1 sets a goal of 50 hours of pro bono work per year for every lawyer [3]. A similar pro bono concept is promoted among architects.

Closer to our own discipline, engineering societies define clear expectations of professional responsibility, including contributing to social good. Voluntary contribution of services is often part of this expectation, and is visible in organizations such as Engineers without Borders [29].

Computing is a younger profession, and expectations for social responsibility and professional service are less clearly defined. However, computing for social good is part of the ACM Code of Ethics and Professional Conduct [7], for example in sections 1.1 and 1.5. Similarly, the Software Engineering Code of Ethics and Professional Practice [42] clearly addresses notions of social responsibility of software engineers including in sections 1.03 and 1.08. Robust CSG-Ed is important to ensure that students understand not only the "do no harm" aspect of professional responsibility, but also the opportunity and responsibility for computing professionals to improve society.

The need for computing students to develop a sense of social responsibility is partially addressed in the ACM Computer Science Curriculum Recommendation Society, Ethics, and Professionalism (SEP) Knowledge Area [21]. The SEP knowledge area and CSG-Ed have a natural synergy where the incorporation of CSG-Ed throughout the curriculum prompts discussion around the question "What is the social responsibility of computing professionals"? These discussions will allow social responsibility to be explored as part of student learning.

CSG-Ed also provides computing education in the authentic computing environment of a visible social issue. An authentic computing environment reflects real-world development and may include industry standard tools and technologies, professional practices, and real stakeholders. Learning within this context helps students develop critical thinking and problem-solving skills. Students learn to think about the root causes of social issues and consider multiple perspectives. It also bridges the gap between theory and practice, between academia and the real-world. This enriched learning environment [15] requires that students design under real-world constraints [5].

Many academic institutions have the promotion of social good as part of their mission. Liberal arts institutions in particular may have humanitarian goals and CSG-Ed is one way for computing departments to support the larger mission of their institutions. Indeed, many institutions have programs, organizations or clubs devoted CS + Social Good including Stanford [71], University of Delaware [75], UNC Chapel Hill [20], Georgia Tech [32], and NYU [60].

CSG-Ed may also help academic institutions fulfill accreditation and professional education requirements. CSG-Ed can provide students with skills required by the current job market. It may also open opportunities for student employment and entrepreneurship in areas such as civic tech and social applications.

As social problems are situated within a particular context and domain, CSG-Ed is a natural vehicle for interdisciplinary collaborations. CSG-Ed allows computing students to understand user needs while also grasping the constraints that context places on solutions. Some natural interdisciplinary areas are computing and sociology where students can work to solve local social problems, and computing and healthcare where students can work to improve various aspects of healthcare.

Another strong reason for providing CSG-Ed is that students are motivated by developing applications and technologies that have positive social impact [9,24,28,48,56,65]. In addition, CSG-Ed affords students the opportunity to become social entrepreneurs and innovators which fosters a culture of innovation and social responsibility within the institution.

While CSG-Ed can increase motivation of all students, there is evidence that CSG-Ed has stronger appeal for underrepresented students [13,65]. In addition, multiple efforts [8,47,61] have shown that providing students with a sense of purpose in their education may increase their sense of belonging, which is known to increase retention of underrepresented groups. Thus, CSG-Ed has the potential to improve computing diversity.

Development of CSG-Ed

The term Computing for the Social Good in Education (CSG-Ed) first appeared in a 2012 ITiCSE Working Group report [36]. Using this event as a dividing point, this section presents a summary of the development of CSG-Ed prior to the 2012 report and then over the years since.

Prior to ITiCSE 2012

As an even earlier ITiCSE Working Group report summarizes [34], the first formal recognition of the need to consider the social implications of computing was in CS1991 [74]. This ACM/IEEE report recommended 11 hours in the CS curriculum be devoted to Social and Professional Context. One direct response to this new curricular recommendation was the US National Science Foundation's funding of ImpactCS; a project to produce detailed instructional resources related to social issues and the impacts of computing [55].

The recommendations in CS1991, ImpactCS, and subsequent work [53] focused on professionalism, computer history, codes of ethics, and intellectual property. It was not until 2000 that the computer science education community considered cultural issues as part of the CS curriculum [54].

It is important to point out that the above recommendations primarily involved a case study approach to incorporating these new topics in the curriculum. A “learn by reading” instead of “learn by doing” approach. However, over the next two decades, experience reports started to appear [1,2,11,16,17,22,25,26,33,34,35,37,49,57,58,64,68]. One favorite was titled "Note to self: make assignments meaningful" [52].

A common denominator in these initial experience reports was that they almost universally were for upper division courses: advanced software engineering, senior capstone courses, HCI electives, etc. Or, as the 2012 ITiCSE Working Group report observed, "only those students who have survived one to two years of duck counting and Checkers will finally get exposed to the potential social value contributions of computing". [36]. This 2012 report is believed to be the first to call for the inclusion of hands-on CSG-Ed activities across the CS curriculum, including the introductory sequence, as well as to provide guidance on how to accomplish this.

Since ITiCSE 2012

The ACM Special Interest Group on Computers and Society (SIGCAS) has long focused on computing ethics (e.g. ACM Code of Ethics, ethics education). In 2014, a newly elected SIGCAS Board added a CSG-Ed agenda. This focus on the integration of CSG-Ed activities throughout the CS curriculum reflected three ideas:

- CSG-Ed provides an important context for introducing computer ethics in the computing curriculum.
- Since computing students work on projects to create software artifacts, why not work on projects that have a positive social value? This achieves the same learning objectives as using toy projects, but with the added benefit of introducing CSG.
- There is significant evidence that suggests CSG-Ed activities can help broaden participation in computing (BPC) and help with retention of underrepresented groups [65].

Starting in 2015, SIGCAS began sponsoring events focusing on how to incorporate CSG-Ed activities across the CS curriculum. These events primarily occurred as pre-conference symposia held in conjunction with the annual ACM Technical Symposium on Computer Science Education (SIGCSE TS). Organized by experienced CSG-Ed practitioners, the goal of these events was to teach a growing set of educators how to become successful CSG-Ed practitioners.

It is difficult to pinpoint an exact date, but approximately six years ago the reputation of the computing industry began to tarnish. The use of social media platforms to subvert free and fair elections; Cambridge Analytica [12] ; large-scale collection and unauthorized use of data that people expected to be private [46]; bias from machine learning applications [23] are a few of the developments that led to this tarnishing. This has led to a significant increase in CSG-Ed interest culminating with the creation of clubs and programs at the institutional level, and the ACM/IEEE/AAAI CS2023's significant focus on the SEP (Society, Ethics, and the Profession) knowledge area.

Approaches to Teaching CSG

There are a variety of ways that CSG-Ed can be incorporated into computing education. This section describes two perspectives for employing CSG-Ed. The first perspective identifies curricular locations for CSG-Ed. The second perspective identifies types of CSG learning experiences.

Curricular Locations

Within a single course - This is perhaps the simplest way to incorporate CSG-Ed. One or more assignments can use a social problem as the application domain (e.g., searching a list of counties with Covid cases to determine the counties with the highest and lowest infection rates). This could be extended to having the entire course revolve around analysis of Covid data.

Across several courses - A CSG theme can be extended across courses. The advantage to this approach is that it reduces the student learning curve for the application domain. For instance, drawing examples and assignments from a campus food pantry software application for use in CS1 (e.g., sorting all items in the food pantry, identify the most popular items in the pantry) is also a natural fit for a CS2 class (e.g., creating a priority queue of students using the pantry).

As a curriculum theme - Students could be introduced to a CSG domain in the first year and build knowledge throughout the program. For instance, supporting a local homeless shelter would provide rich examples for introductory classes (e.g., Reporting on shelter use and resource), and advanced classes (e.g., create a database for shelter management, create a scheduling algorithm for shelter workers).

Course on computing for social good - In this case, the course typically focuses on using computing to address a social problem. These courses may also be interdisciplinary, combining computing students and students from other majors into teams to solve a local social problem [66]

Outside the classroom - Students can participate in short-term experiences such as workshops on CSG, single-day experiences, and hackathons that focus on computing for good such as Civic Hackathons and AI for Social Good Hackathons. Longer-term learning experiences with CSG can be supported via co-ops and internships at non-profits and Non-Government Organizations (NGOs) and clubs that partner to develop applications for their local community such as scheduling software for social services. There are also many humanitarian free and open source (HFOSS) projects that welcome participation by students.

Taxonomy of CSG Learning Activities

CSG learning activities can involve varying levels of engagement by students and effort by instructors [39]. The higher levels also tend to line up with later courses in a curriculum.

Level 1 – Using a CSG context for an existing assignment. (e.g., a sorting assignment that originally utilized a list of sports statistics could be updated to use the number of gun deaths in townships within a state). Level 1 requires the least amount of effort on the part of the instructor since the instructor is already familiar with the assignment deliverables, student effort, and required grading.

Level 2 - Creating new assignments using CSG application domains. Current events can provide fertile ground for these assignments. The assignment may not solve the social problem, but provides students with an understanding of how a specific algorithm, technique, or tool could be used for CSG, (e.g., utilizing public health data to identify risk of nutritional deficiencies for the local population). This level involves a bit more instructor effort than level 1 to identify data and create the assignment.

Level 3 - Address a societal problem. Work can be scoped to be achievable by students while not addressing all aspects of the problem. The goal is to demonstrate a possible solution to the real-world problem, but not to deliver the solution. Students will understand the depth and dimensions of the problem while gaining an understanding of how real-world constraints affect solutions. Level 3 experiences will typically span assignments. (e.g., analyzing recycling effectiveness on campus which could involve conducting a survey, analyzing the results and designing software for effective recycling.) Level 3 can require a substantial effort by the instructor as students are working on a real problem with larger scale and complexity. Learning may need to be supported with multiple assignments, perhaps including learning about the application area.

Level 4 - Working on a societal problem with stakeholders who will use the solution. Students typically interact with potential users, who will be impacted by the proposed solution. Students may be motivated by gaining an understanding of the social need and then taking steps to address that need. (e.g., creating an application to reduce food waste on campus by allowing a campus food service to broadcast availability of food and allow students to place a reservation and pick up that food at a designated location). Level 4 is a larger effort for the instructor since the effort is no longer bounded by the course and interactions with stakeholders must be managed.

Level 5 - Working on a real-world problem with an external software development community already trying to address the need. With this approach students may be motivated by engaging with CSG as well as by developing within a real-world professional environment. Instructor effort can be significant.

It is important to note that the five levels described above need not be approached as isolated endeavors by instructors. In fact, there is good potential to get some economy of scale across instructors and courses if the approach to CSG is shared. For example, if an upper level course engages with a CSG project that has real clients (that is, level 4 or level 5), that effort is likely to provide lots of examples and information that could easily translate to level 1 or 2 CSG activities suitable for courses earlier in the curriculum.

CSG-Ed and the ACM/IEEE-CS/AAAI Curriculum Guidelines for Computer Science (CS 2023)

CS 2023 reflects computing's pervasive impact on modern society and its potential to create both harm and good. This is visible in the inclusion of both the SEP (Society, Ethics, and the Profession) knowledge area and also the SEP knowledge unit in every other knowledge area of the guidelines. This approach reflects the fact that the SEP topics should be integrated into all aspects of computer science. The SEP knowledge area contains eleven knowledge units, six of which are directly related to CSG-Ed: Social Context; Intellectual Property; Privacy and Civil Liberties; Security Policies; Laws and Computer Crimes; and Equity, Diversity and Inclusion. This clear emphasis on SEP in the CS2023 guidelines provides strong motivation for incorporating CSG-Ed in every computing curriculum using some of the approaches outlined in the sections above. CSG-Ed is also a topic in the Curricular Practices effort undertaken in support of CS 2023.

Barriers to Adoption

While there are many benefits to incorporating CSG into higher education, there are also challenges. These barriers exist to some degree in all academic contexts, but they are not insurmountable. The list below summarizes experience of the authors as well as discussions with other instructors about efforts to implement CSG-Ed in various institutions.

Familiarity - Many instructors teach in the manner that they themselves were taught and they were not educated using CSG. Instructors will naturally use approaches and material that have been shown to be successful in their own educational experience. Without this personal experience, instructors may be less likely to incorporate CSG in their classes as they may be unsure of the impact. In addition some instructors may resist incorporating new topics, fearing that it could disrupt the established curriculum or require significant rework.

Instructor time - As computing grows and evolves, there is a need to include more and more material in undergraduate computing curricula. Instructors must already spend significant time to keep course content current. It can be challenging to incorporate new CSG content without impacting other important topics. Also, the addition of CSG into existing courses requires time in order to understand CSG domains and create new assignments, all while maintaining currency of content.

Instructor learning curve - Instructors may lack experience in teaching CSG and may not have sufficient domain knowledge to easily incorporate CSG without spending time researching social good software applications. Instructors may be reluctant to draw examples or assignments from domains with which they are not familiar. In addition, some domains may require simplification in order to find appropriate material and assignments for students. And lastly, instructors may not understand how best to evaluate CSG-Ed assignments as they may require adjustments to grading approaches.

The incorporation of CSG into education may result in highlighting social issues within the classroom which may raise ethical issues on the part of the instructor. Instructors must learn to navigate these issues in order to ensure a safe classroom while addressing a social need. Students may also come from a variety of backgrounds and bring with them varying understanding of social issues. Instructors may need to address this diversity of background when incorporating CSG.

Technology - Some efforts to incorporate CSG into education can face technological barriers. The projects used in CSG may involve complex technologies or tools that are not readily available or accessible to students, requiring additional support and resources. And while features such as faculty development workshops, guest speakers, or community partnerships can enhance the incorporation of CSG into education, academic institutions might not allocate resources for these efforts.

Textbooks - Lastly, there appears to be little support in textbooks for CSG examples. Many instructors use a commercial textbook or courseware and these materials often do not use CSG in their examples. Finding suitable teaching materials and resources that focus on computing for social good can be difficult. This lack of materials increases instructor load as the instructor must then spend additional time to add CSG-related materials.

For all the potential barriers discussed above, the magnitude of the barriers varies depending on what sort of CSG learning activity is introduced. The addition of Level 1 - 3 (described earlier) will require much less instructor effort and have much lower barriers overall. Adding CSG at the levels 4 or 5 will require more careful management of the potential barriers to success.

Examples of CSG-Ed

This section provides a series of examples of ways that CSG has been included in computing education. These examples are possible approaches to engaging students in CSG-Ed.

Course Integration

One easy way to include CSG-Ed is to start with a single assignment within a single course. Many computing assignments are situated within a domain, e.g., calculating the shortest distance to a Starbucks location. These domains could easily be replaced with domains with social impact, e.g., calculating the shortest distance to an emergency room. Such domains can be elicited from the student population. If a local domain of interest has been identified that social domain could be integrated across several different courses, providing a CSG theme within the curriculum. In addition, course integration is an area that is ripe for interdisciplinary collaborations. There may be colleagues in other departments who have social-good problems that need to be solved.

Project: ITiCSE Working Group

Overview: Perhaps the most widely known effort to incorporate computing for social good into the classroom resulted from two ITiCSE working groups [34,36]. The 2010 effort utilized surveys and a set of case studies and provided examples and resources to support instructors interested in incorporating computing for social good into their classes. The 2012 effort provided a framework for adding computing for social good to introductory computing assignments.

Details: These efforts resulted in multiple interesting assignments including:

- The application of the shortest path algorithm to identify the Red Cross office closest to a disaster,
- A GIS application that determines the safety of nuclear plant sites,
- A GIS application that identifies the source location of water pollution,
- Simulation of an emergency and creation of an ad hoc WiFi network to support the response,
- An investigation of the impact on democratic elections of a variety of voting systems

The 2012 effort resulted in a set of resources for instructors new to CSG-Ed or already familiar with CSG-Ed [36].

Curricular Location: The CSG examples focused on introductory computing courses are also applicable across an undergraduate computing curriculum.

Impact: The existence of a framework and resources for supporting CSG-Ed is helpful to CS degree programs across a wide range of academic institutions and environments.

In-House Projects

Adopting or creating a project within the classroom environment allows the instructor to have control over project development while also allowing them to synergize project development with assignments. Projects may be selected from local social organizations on campus, within the local community or within a larger community. Collaborating with on-campus organizations provides students with the opportunity to see the impact of their work within their own community. Students could also be encouraged to propose and lead CSG projects based on their passions, which can increase motivation and engagement.

Project: Campus Plate

Overview: Campus Plate [50,51] is a project created at Baldwin Wallace University (Berea, OH, US) to improve sustainability by reducing food waste while also addressing student food insecurity. The project uses a software application to make surplus food from the campus dining service available to students at no charge.

Details: Campus Plate evolved as a collaboration among multiple groups on campus. The key participants are students and faculty in the Computer Science and Sustainability majors, the campus food service, and a campus community engagement group (The David & Frances Brain Center for Community Engagement). The campus food service provides surplus food that would otherwise be wasted and provides labeling and packaging for individual servings. This food is placed in multiple campus locations that are overseen by student volunteer managers, and availability is broadcast to students who have registered for the Campus Plate application.

The computer science contribution is a system that includes device native iOS and Android apps and a Web application. Students who sign up as potential food recipients may use the iOS or Android apps. The Web app is primarily used for administrative functions.

Curricular Location: Computer Science students have worked on Campus Plate as part of a CS Capstone course. Students also worked on the project as summer interns and as part of an extracurricular group called MOPS, the Mobile and IoT for the Planet and Society Research Group.

Impact: In initial operations, about 80% of the recovered food items are distributed to students rather than going to waste. Students involved with the project gain experience with an authentic, interdisciplinary project with real clients. Students also receive firsthand exposure to a societal issue that is a local concern among students. As one student commented: "I was completely unaware that the number of food-insecure students was that large." [50].

Project: FarmData2

Overview: FarmData2 [14,30] is an application that supports the day-to-day operation of small, diversified vegetable farms. In addition, it facilitates recordkeeping for organic certification and the study of sustainable farming practices. The project was started in the Computer Science department at Dickinson College (Carlisle, PA, US).

Details: FarmData2 is being built in collaboration with the Dickinson College farm and with support from farmOS and PASA Sustainable Agriculture. The farm provides experiential learning for students including education related to renewable energy and sustainable agriculture.

FarmData2 focuses on providing data collection and reporting on farm activities from seeding through harvesting. The features support organic certification and the study of sustainable farming practices. FarmData2 is a plugin module for farmOS which is a Drupal application. The FarmData2 module uses common Web technologies.

Curricular Location: The first computing course introduces students to open source and humanitarian open source projects to exemplify the power of computing to drive positive social change. A sequence of courses at the intermediate level provides experiences with open source tools and processes via FarmData2. These cumulative experiences prepare students for a senior capstone sequence where students engage with a humanitarian or other open source project of their own choosing.

Impact: FarmData2 is the current version of an evolving software application that supports operation of the College farm, the operation of that farm as a certified organic producer, and the collection of data for PASA research studies on sustainable farming practices. FarmData2 provides in-class and internship experiences through which students gain experience with the development of a software product of significant size and the use of tools and processes common in professional software development. Students also experience first-hand the role that computing can play in creating social good.

External Projects

Real-world projects can challenge students to apply their technical skills to social problems that they care about. Projects with stakeholders from different backgrounds will help students to better understand the problem from the stakeholder perspective. Projects in the local community allow students to understand how their efforts can have impact beyond the walls of their institutions. Projects with national or global scope allow students to understand the breadth of influence that computing can have.

Project: GNOME Caribou Onscreen Keyboard

Overview: The GNOME Accessibility team develops applications and a framework that allows people with disabilities to make use of the GNOME desktop, a widely used graphical user interface for Linux. GNOME Accessibility provides assistive technologies for users with impairments and special needs.

Caribou is an application that supports text entry and UI navigation with the goal of providing an input solution for people who access a computer primarily through a switch device. The project desired some modifications to the keyboard, specifically the addition of a backslash and tilde keys to facilitate entry of URLs [27].

Details: Senior-level student software engineering teams developed a keyboard for the Caribou application that included the backslash and tilde keys as well as a "programming keyboard" that included special characters often used by programmers such as the percent sign and parentheses. They developed the requirements, design, implementation and test for the application.

Curricular Location: Software Engineering and Capstone courses that typically involve such project work frequently are found in the final year of an undergraduate degree program when students have gained a reasonable computing background.

Impact: Students were directly involved with the GNOME Accessibility team during project development and reported learning much about professional interactions. They also reported increased motivation from working on a project that directly benefited users with disabilities.

Service Learning

Many academic institutions have a service learning requirement that may allow students to apply computing and project-based learning in organizations addressing social needs.

Project: Centro de Desarrollo y Consultoría Computacional

Overview: At the University of Puerto Rico Rio Piedras Campus after the devastation of Hurricane María in the fall of 2016, the Department of Computer Science developed the Centro de Desarrollo y Consultoría Computacional (CDCC). This consulting center is comprised of professors and students from the University. Via the center, undergraduate students of computer science gain practical experience by participating in computing consulting and development services for the community.

Details: Students created websites, mobile applications and databases for the university, local non-profits, and businesses. For example, they created an evaluation app for the biology department to measure the growth in scientific and computing identity of student participants in the Interdisciplinary and Quantitative Biology Research Experience for Undergraduates (IQ BIO REU).

Curricular Location: Students were recruited from the Software Engineering class where they solved similar problems with local clients. If the client liked the solution, the students were hired to implement it through the CCDC.

Impact: These service experiences met a need in the community while building the professional and programming skills of students in the department. In addition, more students were hired to work on local community projects through the CCDC.

Project: Berea College Active Learning and Service Learning Courses

Overview: All Berea College students get an Active Learning Experience, which might be a service-learning course or an internship. In this program, service-learning staff work with faculty and community organizations to address community needs and guide students in structured reflections.

Details: Community-based service-learning projects are incorporated at several points in the CS curriculum at Berea including CS0, Database Systems, and Open Source Software Engineering courses [72]. Projects have included Websites for community organizations, databases for a local food bank, and contributions to the Runestone Academy interactive textbook hosting application. There is also a Student Software Development Team that works on IT solutions for campus organizations [45,81].

Curricular Location: Computer science students are afforded the opportunity to participate in active learning courses throughout all of their years in college.

Impact: Students consider the active-learning component of their education to be very valuable. Students gain an understanding of the importance of civic engagement while deepening their understanding of course content. Students who take a CS0-level course with service learning are more likely to continue on in a computing degree and members of the Student Software Development Team find they are much better prepared for work with large software systems.

External Activities

There are multiple opportunities beyond the classroom where students can engage in CSG activities. Some academic institutions have clubs such as Harvard's Tech for Social Good [44]. Techshift [73] is an alliance of CS+Social Good clubs that works to expand the network of organizations that promote social good and also hosts an annual summit.

Hackathons that focus on social needs may be motivating. Code for Good [18], Women Who Code's Hackathon for Social Good [82], and Hacks for Humanity [43] are hackathons that focus on developing technology solutions for social good. These events bring together students, professionals, and mentors to collaborate and create prototypes that address pressing social issues. Coding internships such as bootcamps and summer experiences may be designed for social good. Many of Google's Summer of Code experiences are aimed at social good projects.

Research projects may be student research projects that leverage computing for social good in which students may participate either via a paid position or internship. Several institutions have research groups organized around social good such as University of Illinois Urbana-Champaign, School of Information Sciences [80] and the University of Florida Computing for Social Good Lab [19].

Throughout the Curriculum

Project: British Secondary Education

Overview: In England, computing education starts at age 5 and continues through secondary education. Guidelines for these years include social good considerations. For example, at the secondary level, there is national guidance [78] that specifies students must "understand the impacts of digital technology to the individual and to wider society".

Curricular Location: University curricula are defined by individual institutions. However, the subject content is informed by national subject benchmarks. The Computing benchmark recognizes the potential impact of computer science, and that "computing courses should address the sustainable development agenda through highlighting key economic, social and environmental issues that the discipline influences, whether positively or negatively" [67].

The British Computer Society – the chartered institute for IT – represents the discipline, nationally and internationally. One of the aims of the BCS is to "make IT good for society" [10]. It provides accreditation of courses based on their subject coverage, and includes requirements that accredited courses will ensure graduates "design solutions for complex problems that meet a combination of societal, user, business and customer needs as appropriate. This will involve consideration of applicable health and safety, diversity, inclusion, cultural, societal, environmental and commercial matters, codes of practice and industry standards."

Impact: While an appreciation of the ethical dilemmas does not mean someone will necessarily choose to do good, the consideration of legal, social, ethical and professional responsibilities and duties can offer opportunities to explore what good may mean, and how computing can be used for good. The United Nations Sustainable Development Goals provide a set of examples and challenges that can further illustrate this, and these can be combined as a framework to encourage computing students to develop the relevant competencies, and consider how they can potentially do good [41].

Best Practices for Implementing CSG-Ed

In this section we describe a variety of ways to start or further CSG-Ed as well as addressing ways to promote CSG-Ed.

Approaches to Incorporating CSG

Connect with Real-World Problems: A natural entrance to CSG-Ed is to determine the social problems that motivate your students. Students may be motivated by the possibility of solving problems that impact them directly or by being able to have impact on local or global problems. Tailoring assignments to align with students' interests and motivations makes the content more relevant and engaging. Polling students for their interests in social problems may help identify domains that are of specific interest to students. Projects and assignments can then be selected that address social issues or benefit local communities, demonstrating the practical applications of computing.

One simple assignment to connect students with real-world problems is a case study of a social problem that has been improved by computing. Another assignment is to set up a debate and discussion on a topic of social need and possible solutions that will highlight the intricacies in solving social issues while promoting critical thinking. Assignments that require students to explore the root causes of social problems and consider multiple perspectives promote critical thinking and heighten students' awareness of the power of computing.

Engage Locally: The ability to address social problems with local impact may motivate students more strongly. Involving local organizations provides learning in real-world conditions and real-world impact. Collaboration with other disciplines outside of computing such as sociology or environmental studies also provides opportunities for students to gain a holistic perspective on social issues and fosters interdisciplinary problem-solving skills.

Assignments to get started in this area include field trips to local organizations that address a social need in order to gain a deeper understanding of the social need and possible solutions. This could be combined with a reflective journal or essay where students provide observations on the ability of computing to address a social need. In addition, guest speakers from local organizations that work on social problems could be invited to share their experiences and insights with students and speak about their computing needs.

Structure Learning with Objectives and Assessment: When creating a class that incorporates CSG, consider the specific learning objectives related to social impact. Develop assessment methods that evaluate both technical skills and the impact of computing solutions on social issues. Provide constructive feedback.

Vote For Open: Open source software, open data, open hardware, open government and other "open" venues provide an environment abundant with learning opportunities. Computing in the open encourages collaboration, transparency, and innovation in solving social issues. Open resources that are free and open source software can be customized to fit a local need. In addition, student contributions to an open project are publicly visible, providing evidence of professional ability to potential employers.

Leverage Existing Resources: Existing teaching materials and resources can be leveraged to reduce the learning curve for instructors. Make use of existing teaching resources and contribute new materials to them.

Create Inclusive Community: When addressing social issues in the classroom, it is especially important to create an inclusive and collaborative environment where students feel safe to discuss social issues and propose solutions.

Connecting CSG-Ed to Computing Ethics Education

Computing ethics has been a recommended part of computing education for many years. However, while the study of ethics within computing is well established, coverage can be fragmentary and variable in practice. Ethics may be taught by computing staff, or as a service course by another department. Ethics is identified as required in various guidance and syllabus documents, and has its own entry within the ACM curricula guidelines, and specific national ones such as the UK Computing benchmark. Codes of professional practice, such as IEEE, BCS and ACM typically mention ethics and the need for professionals to act in an ethical way.

CSG-Ed can play a supportive role in computing ethics education. Discussion of computing ethics usually includes computing that is intentionally harmful such as cybercrime, and also unintended harm such as algorithmic bias. CSG-Ed provides an excellent context to introduce issues of professional responsibility and the positive impact of computing. While a computing professional can behave ethically without ever participating in CSG, most codes of ethics encourage practitioners to include social good as part of their sense of professional responsibility. CSG-Ed provides great examples for introducing students to this type of professional behavior to produce positive impact.

Conclusion

Given recent global events, providing students with an understanding of the potential of computing for social good is growing in importance. Opportunities should be investigated to expand CSG-Ed while preparing students for a rapidly changing digital landscape.

Computing continues to evolve rapidly and to have amazing impacts on modern life. In fact, it is difficult to identify areas of human activity that are not increasingly intertwined with computing. In this context, the need for computing education to emphasize professional responsibility is critical. Computing students must have solid grounding both in the need to avoid harm by pursuing socially responsible computing and to create public benefit via computing for social good. The ACM Code of Ethics and Professional Responsibility has recognized this need for many years. Computing educators have a pivotal role in ensuring that students are well-prepared to meet this challenge.

References

1. Aheer, K. Bauer, K. and Macdonell, A.C. Internationalizing the Student Experience Through Computing for Social Good. in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. Association for Computing Machinery, New York, NY, USA, 434–440. (2020). https://doi.org/10.1145/3328778.3366853

2. Aheer, K. and Macdonell, A.C. 2021. Hello, world: an internalization at home project for computing for social good. ACM SIGCAS 49, 1 (February 2020), 13–14. https://doi.org/10.1145/3447892.3447896

3. American Bar Association. ABA Model Rule 6.1. 2023. Accessed December 19, 2023. https://www.americanbar.org/groups/probono_public_service/policy/aba_model_rule_6_1/

4. American Medical Association. AMA Declaration of Professional Responsibility. 2023. Accessed December 19, 2023. https://www.ama-assn.org/delivering-care/public-health/ama-declaration-professional-responsibility

5. Anderson, R.J. Anderson, R.E. Borriello, G. and Pal, J. An Approach to Integrating ICTD Projects Into an Undergraduate Curriculum. *Proceedings, 41st ACM Technical Symposium on Computer Science Education*. 529–533, ACM: New York, NY, USA, 2010.

6. Anderson, J. You Can't Code for Humanity: AI, Algorithms, and the Bias of Machine Learning. *Resources for Gender and Women's Studies* 44.1/2 (2023). 18-19.

7. Association of Computing Machinery. https://www.acm.org/code-of-ethics. Accessed December 19, 2023.

8. Banerjee, S. and Mazur, N. Service Learning in Computing: Creating Computer Science Pipeline by Attracting and Engaging High School Students. *J. of Computer Science in Colleges*. 33, 6 (June 2018), 173-175.

9. Bates, R. et al. A Project-Based Curriculum for Computer Science Situated to Serve Under-represented Populations. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education*. Association for Computing Machinery, New York, NY, USA. 2022. 585–591. https://doi.org/10.1145/3478431.3499312

10. BCS. Making IT good for society https://www.bcs.org/articles-opinion-and-research/making-it-good-for-society/ 2015 Accessed April 25, 2024.

11. Beck, R.E. Computing for the social good: Experiences outside the walls. *Computers and Society*. ACM SIGCAS 45(2):43–43, July 2015.

12. Berghel, H. Malice Domestic: The Cambridge Analytica Dystopia. in *Computer*, 51(5), 84-89, May 2018, doi: 10.1109/MC.2018.2381135.

13. Braught, G and Siddiqui, F. Factors Affecting Project Selection in an Open Source Capstone. In *Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education*. 2022. Association for Computing Machinery, New York, NY, USA, 358–364. https://doi.org/10.1145/3502718.3524760

14. Braught, G. Huss-Lederman, S. Jackson, S. Turner, W. and Wurst, K.R. Engagement Models in Education-Oriented H/FOSS Projects. in *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (2023). Association for Computing Machinery, New York, NY, USA, 409–415. https://doi.org/10.1145/3545945.3569835

15. Buckley, M. et al. Benefits of Using Socially Relevant Projects in Computer Science and Engineering Education. In *Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education*, 2004, 482–486, ACM: New York, NY.

16. Buckley, M. Nordlinger, J. and Subramanian, D. Socially relevant computing. *In Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*. 347–351. ACM: New York, NY. 2008.

17. Cai, Y. Integrating sustainability into Undergraduate Computing Education. in *Proceedings of the 41st ACM Technical Symposium on Computer Science Education*. 524–528. ACM: New York, NY. 2010.

18. Code for Good. https://careers.jpmorgan.com/us/en/students/programs/code-for-good. Accessed 26 Nov. 2023.

19. Computing for Social Good Lab. https://www.computingforsocialgood.org/. Accessed 26 Nov. 2023

20. CS + Social Good at UNC Chapel Hill. https://cssgunc.org/. Accessed 26 Nov. 2023.

21. CS2023. ACM/IEEE-CS/AAAI Computer Science Curricula. https://csed.acm.org/, Accessed 26 Nov. 2023.

22. Connolly, R.W. Beyond good and evil impacts: rethinking the social issues components in our computing curricula. in *Proceedings of the 16th Annual Joint Conference on*

Innovation and Technology in Computer Science Education. 228–232. ACM: New York, NY. 2011.

23. Dastin, J. Amazon Scraps Secret AI Recruiting Tool that Showed Bias Against Women. *Reuters Business News*. 2018 October 10.
24. Dahlberg, T. Barnes, T. Buch, K. Bean, K. Applying Service Learning to Computer Science: Attracting and Engaging Under-Represented Students. *Computer Science Education*. 20,3, 169-180. 2010.
25. Dekhtyar, A. et al. From RE Cares to SE Cares: Software Engineering for Social Good, One Venue at a Time. in *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering in Society*. Association for Computing Machinery, New York, NY. 49–52. 2020. https://doi.org/10.1145/3377815.3381373
26. Ellis, H.J.C. Morelli, R.A. de Lanerolle, T. and Hislop. G.W. Holistic Software Engineering Education Based on a Humanitarian Open Source Project. in *Proceedings of the 20th Conference on Software Engineering Education & Training*. IEEE Computer Society, Washington, DC, USA, 327-335. 2007. DOI=10.1109/CSEET.2007.26
27. Ellis, H.J.C. Chua, M. Jadud, M.C. and Hislop, G.W. Learning Through Open Source Participation. in *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education*. Association for Computing Machinery, New York, NY, USA, 83–84. 2011. https://doi.org/10.1145/1953163.1953191
28. Ellis, H.J.C. Hislop, G.W. Jackson, S. and Postner, L. Team Project Experiences in Humanitarian Free and Open Source Software (HFOSS). *ACM Transactions on Computer Education*. 15, 4. December 2015. https://doi.org/10.1145/2684812
29. Engineers without Borders International. 2023. https://www.ewb-international.org/ Accessed 2023 December 19.
30. FarmData2. https://github.com/DickinsonCollege/FarmData2. Accessed 2023 Nov. 26.
31. Forbes. How Target Figured Out a Teen Girl Was Pregnant Before Her Father Did. 2012 Feb 16. Accessed 2023 Jan. 18.
32. GT CS + Social Good. https://gtcssocialgood.com/. Accessed 2023 Nov. 26.
33. Goadrich, M. et al. Civic Engagement Across the Computing Curriculum. in *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*. Association for Computing Machinery, New York, NY. 649–650. 2019. https://doi.org/10.1145/3287324.3287335.
34. Goldweber, M. et al. Enhancing the Social Issues Components in Our Computing Curriculum: Computing for The Social Good. In *Proceedings of the 15th Annual Joint Conference on Innovation and Technology in Computer Science Education*. June 2010.
35. Goldweber, M. Computing for the Social Good: A Service Learning Project. in *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education*. 379–379, New York, NY, USA, 2011. ACM.
36. Goldweber, M. et al. A Framework for Enhancing the Social Good in Computing Education: A Values Approach. In *Proceedings of the Final Reports on Innovation and*

Technology in Computer Science Education 2012 Working Groups. Association for Computing Machinery, New York, NY, 16–38. 2012. https://doi.org/10.1145/2426636.2426639.

37. Goldweber, M. Computer Science Education for Social Good. *Computers and Society.*, 45(2):29–30, July 2015.
38. Goldweber, M. Strategies for Adopting CSG-Ed in CS 1. In Proceedings of the 3rd Conference for Research on Equity and Sustained Participation in Engineering, Computing, and Technology. 2018.
39. Goldweber, M, Kaczmarczyk, L. and Blumenthal, R. Computing for the Social Good in Education. *ACM Inroads*.10,4. December 2019,
40. Goldweber, M. et al. A Hands-On Tutorial on How to Incorporate Computing for Social Good in The Introductory Course Sequence. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V.2*, 1039–1040, ACM: New York, NY, 2022.
41. Gordon, N. Sustainable Development as a Framework for Ethics and Skills in Higher Education Computing Courses. In *Integrative Approaches to Sustainable Development at the University Level*. 345-357. Springer. 2015.
42. Gotterbarn, D. Miller, K. and Rogerson, S. Software Engineering Code of Ethics. *Communications of the ACM*. 40, 11. 110-118. November 1997. DOI: 10.1145/265684.265699
43. Hacks for Humanity. https://www.hacksforhumanity.io/. Accessed 2023 Nov. 26.
44. Harvard Computer Society Tech for Social Good. https://socialgood.hcs.harvard.edu/. Accessed 2023 Nov 26.
45. Heggen, S. and Myers, C. Hiring Millennial Students as Software Engineers. In *Proceedings, The Second Annual IEEE/ACM International Workshop on Software Engineering Education for Millennials*. Gothenburg, Sweden. June 2018.
46. Hill, K. How Target Figured Out a Teen Girl Was Pregnant Before Her Father Did. *Forbes Magazine*. Feb. 2012.
47. Hoffman, B. Morelli, R. and Rosato, J. Student Engagement is Key to Broadening Participation in CS. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*. ACM, New York, NY, 1123-1129. 2019. DOI: https://doi.org/10.1145/3287324.3287438
48. Isvik, A. Lytle, N. Catete, V. and Barnes, T. Characterizing High School Participants' Motivations and Outcomes in a Service-Oriented Summer Internship. In *Proceedings of the 22nd Koli Calling International Conference on Computing Education Research*. Association for Computing Machinery, New York, NY. 1–12. https://doi.org/10.1145/3564721.3564734
49. Khan, N.Z. and Luxton-Reilly, A. Is computing for Social Good the Solution to Closing the Gender Gap in Computer Science? In *Proceedings of the Australasian Computer Science Week Multiconference*. 1–5, ACM: New York, NY, USA. 2016.

50. Krupp, B. Gersey, J. Lebo, F. Campus Plate: Connecting Students on College Campuses to Reduce Food Waste and Food Insecurity. *2022 International Conference on Research in Adaptive and Convergent Systems*. ACM: New York, NY, USA.

51. Krupp, B. Lebo, F. Note: Campus Plate: Reducing Food Waste and Food Insecurity on College Campuses using Smartphones. *ACM Computing and Sustainable Societies*. 2022.

52. Layman, L. Williams, L. and Slaten, K. Note to Self: Make Assignments Meaningful. In *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*. Association for Computing Machinery, New York, NY. 459–463. 2007. https://doi.org/10.1145/1227310.1227466

53. Little, J.C. et al. Integrating Professionalism and Workplace Issues into the Computing and Information Technology Curriculum: Report of the ITiCSE'99, Working Group on Professionalism. *SIGCSE Bulletin*. 31, 4. 106–120. 1999. https://doi.org/10.1145/349522.571917

54. Little, J.C. et al. Integrating Cultural Issues into The Computer and Information Technology Curriculum. In Working Group Report from ITiCSE on Innovation and Technology in Computer Science Education. 136–154. 2001. Association for Computing Machinery, New York, NY. https://doi.org/10.1145/571968.571975

55. Martin, C.D. et al. A framework for Implementing and Teaching the Social and Ethical Impact of Computing. *Education Information Technology* 1, 101–122. 1996. https://doi.org/10.1007/BF00168276.

56. McAlear, F. et al. Do Social and Emotional Learning Outcomes and Instructional Practices Promote Persistence in Computer Science for Underrepresented Secondary Students of Color? In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. ACM: New York, NY, USA, 1101-1101. 2018. DOI: https://doi.org/10.1145/3159450.3162235

57. Mertz, J. Computing for the Social Good and Cultivating Cultures for Ethical Computing. *Computers and Society*. 45(2):39–40, July 2015.

58. Morelli, R. et al. Revitalizing Computing Education Through Free and Open Source Software for Humanity. *Communication of the ACM*. 52, 8. 67-75. August 2009. DOI=10.1145/1536616.1536635

59. Morgan, B. Ellis, H.J.C. and Hislop, G.W. "Student Software Engineering Learning in HFOSS Projects". In *Proceedings, ASEE Annual Conference & Exposition*. 2019. ASEE Conferences. https://peer.asee.org/33300 Accessed 2023 Dec. 01.

60. NYU CS + Social Good. https://vasans.hosting.nyu.edu/cssg/. Accessed 2023 Nov. 26.

61. Ordóñez, P. Krishnaswam, K. Tull, R.G. and Goldberg, M. Assistive Technology Research as a Mechanism to Broaden the Participation of Women, Underrepresented Minorities, and Persons with Disabilities. In *Latin American and the Caribbean Conference of Engineering and Technology*. 2014

62. Pearce, J.L. and Nakazawa, M. The funnel that Grew our CIS Major in the CS Desert. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*. 503-507. 2008.

63. Pearce, J.L. Requiring Outreach from A CS0-Level Robotics Course. *The Journal of Computing Sciences in Colleges.* 205. 2011.

64. Postner, L. Using Humanitarian Free and Open Source Software (HFOSS) to Introduce Computing for The Social Good. Computers and Society. 45(2):35–35. 2015. Association for Computing Machinery, New York, NY

65. Postner, L. Hislop, G.W. and Ellis, H.J.C. Humanitarian Applications Increase Interest and Motivation of Women in Computing. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*. Association for Computing Machinery, New York, NY, USA, 416–422. 2023. https://doi.org/10.1145/3545945.3569832.

66. Pulimood, S.M. Pearson, K. and Bates, D. Encouraging CS Students to Compute for Social Good Through Collaborative, Community-Engaged Projects. *Computers and Society*. 49, 1 2020. 21–22. Association for Computing Machinery, New York, NY. https://doi.org/10.1145/3447892.3447900.

67. QAA. Subject Benchmark Statement: Computing. https://www.qaa.ac.uk/the-quality-code/subject-benchmark-statements/computing. Accessed 2023 Nov 26.

68. Rader, C. Hakkarinen, D. Moskal, B. and Hellman, K. Exploring the Appeal of Socially Relevant Computing: Are Students Interested in Socially Relevant Problems? In *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education*. 423–428, ACM: New York, NY. 2011.

69. Shneiderman, B. Computer Science Education and Social Relevance. SIGCSE Bulletin. 3(1):21–24, March 1971. Association for Computing Machinery, New York, NY.

70. ACM Computers and Society. https://www.sigcas.org/tag/csg-ed/. Accessed 26 Nov. 2023.

71. Stanford Cardinal Service. https://cardinalservice.stanford.edu/opportunities/cs-social-good. Accessed 26 Nov. 2023.

72. Stone, J.A. et al. Community-based projects for Computing Majors: Opportunities, Challenges and Best Practices. In Proceedings of the 43rd ACM technical symposium on Computer Science Education (SIGCSE '12). Association for Computing Machinery, New York, NY, USA, 85–86. https://doi.org/10.1145/2157136.2157166"

73. TechShift. https://www.techshift.org/. Accessed 2023 Nov. 26.

74. Tucker, A.B., Ed. Computing Curricula 1991. ACM: New York, NY. https://doi.org/10.1145/103701.103710

75. CS+Social Good University of Delaware. https://studentcentral.udel.edu/organization/cssocialgood. Accessed 2023 Nov. 26.

76. UK Government. National curriculum in England: Computing Programmes of Study https://www.gov.uk/government/publications/national-curriculum-in-england-

computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study#key-stage-1 Accessed 2023 Nov. 26.

77. UK Department for Education. A-level Computer Science Subject Content https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/302105/A_level_computer_science_subject_content.pdf Accessed 2023 Nov. 26.
78. UK Department for Education. Computer Science GCSE Subject Content. https://assets.publishing.service.gov.uk/media/5a7e3cb440f0b62305b81b02/Computer_Science_GCSE_-_subject_content_-_final.pdf. Accessed 26 Nov. 2023.
79. UK Department for Education. Computer science: GCSE Subject Content. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/397550/GCSE_subject_content_for_computer_science.pdf Accessed 26 Nov. 2023.
80. University of Illinois Urbana-Champaign School of Information Sciences, Computing for Social Good. https://ischool.illinois.edu/research/areas/computing-social-good. Accessed 26 Nov. 2023.
81. Williams, B. Cruz, G. and Heggen, S. A Student-Based Software Development Team and Their Response to COVID-19. *Journal of Computing Sciences in Colleges* 36,5 (2021): 220-229.
82. WWCode. Hackathon for Social Good. https://www.womenwhocode.com/hackathon-social-good/. Accessed 26 Nov. 2023.
83. Zin, P. Heggen, S. Pauly, J. Sustainability Dashboard Management System. *Journal of Technologies in Society*. 12,3-4, 1-9. 2016.

Author Details

Heidi J.C. Ellis
Western New England University
1215 Wilbraham Rd
Springfield, MA 01119
Ellis@wne.edu

Gregory W. Hislop
Drexel University
3675 Market St.
Philadelphia, PA 19104
Hislop@drexel.edu

Mikey Goldweber
Denison University
100 West College Street
Granville, Ohio 43023
goldweberm@denison.edu

Samuel Rebelsky
Grinnell College

1115 8th Avenue
Grinnell, IA 50112
rebelsky@grinnell.edu

Janice Pearce
Berea College
101 Chestnut St.
Berea, KY 40404
jan_pearce@berea.edu

Patti Ordonez
University of Maryland Baltimore County
1000 Hilltop Circle
Baltimore, MD 21250
patti.ordonez@umbc.edu

Marcelo Pias
Federal University of Rio Grande
Rio Grande do Sul, Brazil
mpias@furg.br

Neil Gordon
University of Hull
Cottingham Road
Hull, HU6 7RX
N.A.Gordon@hull.ac.uk