

How We Did It: Integrating Inclusive Design across the Undergraduate Computer Science Curriculum

Patricia Morreale
pmorreale@kean.edu
Kean University
Union, NJ, USA

Kyle J. Harms
kyle.harms@cornell.edu
Cornell University
Ithaca, NY, USA

Margaret Burnett
burnett@eecs.oregonstate.edu
Oregon State University
Corvallis, Oregon, USA

Daehan Kwak
dkwak@kean.edu
Kean University
Union, NJ, USA

ABSTRACT

Inclusive design appears rarely, if at all, in most undergraduate computer science (CS) curricula. As a result, many CS students graduate without knowing how to apply inclusive design to the software they build and go on to careers that perpetuate the proliferation of software that excludes communities of users. Our panel of CS faculty will explain how we have been working to address this problem. For the past several years, we have been integrating bits of inclusive design into multiple courses in CS undergraduate programs, which has had very positive impacts on students' ratings of their instructors, students' ratings of the education climate, and students' retention. The panel's content will include mostly concrete examples of how we are doing this, so that attendees can leave with an in-the-trenches understanding of what this looks like for CS faculty across specialization areas and classes. We also show how it can be used in a department's BPC Plan and point to resources on the CRA's BPCnet Activity Library and on OERcommons, to enable interested faculty to move forward with this approach in their own classes and departments.

CCS CONCEPTS

• **Human-centered computing**; • **Applied computing** → **Education**;

KEYWORDS

Inclusive design, computer science education, pedagogy

ACM Reference Format:

Patricia Morreale, Margaret Burnett, Kyle J. Harms, and Daehan Kwak. 2025. How We Did It: Integrating Inclusive Design across the Undergraduate Computer Science Curriculum. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE TS 2025)*, February 26-March 1, 2025, Pittsburgh, PA, USA. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3641555.3704715>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
SIGCSE TS 2025, February 26-March 1, 2025, Pittsburgh, PA, USA
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0532-8/25/02.
<https://doi.org/10.1145/3641555.3704715>

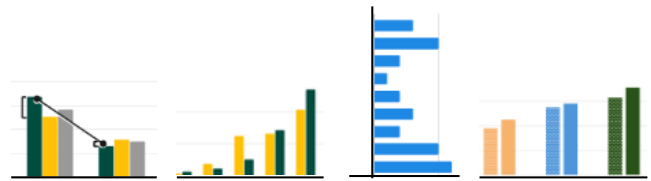


Figure 1: Thumbnails of prior results (Left to right): Lower failures/ withdrawals/ incompletes [5]; Higher ratings of faculty [5]; Positive changes in education climate [5]; Student-built software more inclusive than in prior years [6].

1 SUMMARY

The news media abounds with articles about software that is inequitable, misleading, and biased. But why should we expect it to be otherwise? Most CS education curricula do little to educate CS students on how to avoid creating software with such properties. Even in CS departments offering some form of ethical/responsible/inclusive CS, the offerings are usually isolated courses or assignments, segregated from “regular” CS offerings. As CS education scholars like Gray et al. and Oleson et al. have shown, such isolated efforts do not equip students to know how to bring what little they have learned about such concepts into the way they build software [7, 11].

The CS faculty on this panel have been trying out a different approach in two ways—an across-the-curriculum, systemic version that integrates inclusive design into the mainstream CS work across all four years of the undergraduate curriculum, and a less extensive way that does so in just one or a few classes. Both variants aim (1) to provide CS students with actionable skills they need to avoid these problems, and with (2) to give them practice using these skills in their mainstream CS work, and (3) to displace very little course content at the same time. The faculty’s efforts in both versions have resulted in improvements in students’ retention, students’ ratings of their instructors, students’ feelings of belonging, and students’ practices in designing their own software for inclusion [5, 6, 8]. Their efforts have also broadened participation by minoritized populations [5]. Figure 1 depicts some of these results.

In this panel, faculty from Kean University, Cornell University, and Oregon State University will share in-the-trenches examples and stories of what they did in their classes to ultimately bring about

these results. Because the activities they devised for their classes are based on the InclusiveMag family [9] of inclusive design methods (e.g., GenderMag [4] and the emerging SocioeconomicMag [1, 3]), the panel will begin with the moderator briefly explaining how InclusiveMag methods work. Following this background, the CS faculty will present concrete examples of what they did and why, including a CS Chair who will provide a department chair’s perspective. Our goal is to provide undergraduate CS faculty and Chairs with concrete information and reusable teaching materials, to enable them to use this approach to both broaden participation in CS majors and graduate more inclusive software professionals, one CS class at a time.

2 PANEL STRUCTURE

Introduction (7 minutes): The panel moderator will begin by (1) describing the motivation for integrating inclusive design concepts into undergraduate computer science education; (2) briefly introducing the InclusiveMag family; and (3) summarizing results of teaching it. Then, the moderator will introduce the panelists and moderate the discussion.

Panelist Statements (10 minutes each = 30 minutes total): Each panelist will describe their approaches and practices for incorporating inclusive design methods into undergraduate CS courses, as well as their experiences from doing so.

Discussion (30–35 minutes): The moderator will welcome questions from the in-person and Zoom audiences. In addition, to both seed the discussion and help integrate hybrid participants, we will host a Google form (<https://forms.gle/Ht6RAZRomwmz1Q1B7>) to start collecting questions in advance.

We hope interested audience members will be able to identify some actionable solutions they can use in their own settings, either for their courses or for systemic, across-curriculum changes.

3 POSITION STATEMENTS

Margaret Burnett (Moderator) is a Professor at Oregon State University, and co-leads the team that created GenderMag and SocioeconomicMag. She also initiated the application of these methods to improving CS Education’s inclusivity. After introducing the approach, she will turn to the panelists, so they can show how they used the approach, made it better, and made it work in their classrooms.

Kyle Harms (Panelist) is a Senior Lecturer of Information Science at Cornell University. Kyle’s experience with GenderMag involves the development of a “mini-curriculum” for an online web design and programming course, and incorporating the GenderMag method into a flipped web programming course of 100–300 students. He will note the challenges of teaching GenderMag at scale and share active-learning in-class activities for evaluating and redesigning websites for inclusivity.

Daehan Kwak (Panelist) is an Associate Professor of Computer Science at Kean University, where he has integrated both GenderMag and SocioeconomicMag (SESMag) into CS0. His curriculum changes to integrate GenderMag and SESMag into his classes have been modest, yet students’ understanding and engagement improvements have been significant. He will show two concrete examples

of how he has done this in CS0, and will talk about how his integrations have played out in his classrooms.

Patricia Morreale (Panelist) is a Professor and Chair of Computer Science and Technology at Kean University, a public urban Hispanic Serving Institution (HSI). Since 2015, she has been using Inclusive Design methods in Software Engineering and Human-Computer Interaction courses, but her remarks will focus on her Chair activities. In 2020, she worked with other department faculty to introduce and reinforce inclusive design in each year of the 4-year undergraduate CS and IT programs [12]. She will explain how she engaged Kean’s CS faculty to explore this approach across the curriculum, and how it can be used in a department-wide BPC Plan (see also [2]).

More examples: Depending on the audience’s Q&A, the panel will also be prepared to show examples from other courses—a Data Structures course, a beginning Electrical Engineering course, a Database course, and a Capstone course.

Resources: Our resources on BPCNet [2] and OERCommons [10].

ACKNOWLEDGMENTS

This work was supported in part by NSF awards IIS-1901031, DUE-2042324, DUE-2042341, EES-1928452, and EES-2345334, and by award 2021-67021-35344 jointly sponsored by USDA/NIFA and NSF.

REFERENCES

- [1] Puja Agarwal, Divya Prem, Christopher Bogart, Abrar Fallatah, Aileen Abril Castro-Guzman, Pannapat Chanpaisaeng, Stella Doehring, Margaret Burnett, and Anita Sarma. 2023. SocioEconomicMag Meets a Platform for SES-Diverse College Students: A Case Study. *arXiv preprint arXiv:2304.04873* (2023).
- [2] BPCnet Resource Portal. 2023. BPCNet Resources / GenderMag-Teach Resources. <https://bpcnet.org/resources/gendermag-teach/>
- [3] Margaret Burnett, Abrar Fallatah, Catherine Hu, Christopher Perdriau, Christopher Mendez, Caroline Gao, and Anita Sarma. 2024. Toward an Actionable Socioeconomic-Aware HCI. *arXiv preprint* (2024).
- [4] Margaret Burnett, Simone Stumpf, Jamie Macbeth, Stephann Makri, Laura Beckwith, Irwin Kwan, Anicia Peters, and William Jernigan. 2016. GenderMag: A method for evaluating software’s gender inclusiveness. *Interacting with Computers* 28, 6 (2016), 760–787.
- [5] Rosalinda Garcia, Patricia Morreale, Lara Letaw, Amreeta Chatterjee, Pankati Patel, Sarah Yang, Isaac Tijerina Escobar, Geraldine Jimena Noa, and Margaret Burnett. 2023. “Regular” CS × inclusive design = smarter students and greater diversity. *ACM Trans. Comput. Educ.* 23, 3, Article 34 (Jul 2023), 35 pages. <https://doi.org/10.1145/3603535>
- [6] Rosalinda Garcia, Patricia Morreale, Pankati Patel, Jimena Noa Guevara, Dahana Moz-Ruiz, Sabyatha Satish Kumar, Prisha Velhal, Alec Busteded, and Margaret Burnett. 2024. Beyond “awareness”: If we teach inclusive design, will students act on it?. In *ACM Conference on International Computing Education Research*.
- [7] Colin M Gray, Shruthi Sai Chivukula, Cassandra Melkey, and Rhea Manocha. 2021. Understanding “dark” design roles in computing education. In *ACM Conference on International Computing Education Research*. 225–238.
- [8] Lara Letaw, Rosalinda Garcia, Heather Garcia, Christopher Perdriau, and Margaret Burnett. 2021. Changing the online climate via the online students: Effects of three curricular interventions on online CS students’ inclusivity. In *ACM Conference on International Computing Education Research*. 42–59.
- [9] Christopher Mendez, Lara Letaw, Margaret Burnett, Simone Stumpf, Anita Sarma, and Claudia Hilderbrand. 2019. From GenderMag to InclusiveMag: An inclusive design meta-method. In *2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. IEEE, 97–106.
- [10] OERCommons Site. 2024. GenderMag-Teach+InclusiveMag. <https://oercommons.org/groups/gendermag-teach-inclusivemag/10149/>
- [11] Alannah Oleson, Meron Solomon, and Amy J Ko. 2020. Computing students’ learning difficulties in HCI education. In *ACM CHI Conference on Human Factors in Computing Systems*. 14 pages.
- [12] Pankati Patel, Dahana Moz-Ruiz, Rosalinda Garcia, Amreeta Chatterjee, Patricia Morreale, and Margaret Burnett. 2024. From workshops to classrooms: Faculty experiences with implementing inclusive design principles. In *ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2024)*. 1035–1041. <https://doi.org/10.1145/3626252.3630861>