REQUIREMENTS

Editor: **Sarah Gregory**
Intel Corporation
sarah.c.gregory@ieee.org

Gender Inclusivity as a Quality Requirement: Practices and Pitfalls

Mariam Guizani, Lara Letaw, Margaret Burnett, and Anita Sarma

GENDER INCLUSIVITY IN software is gaining attention from researchers and practitioners, with some seeing it as a nonfunctional requirement. To investigate how gender inclusivity can be incorporated into creating software, we gathered data during periods ranging from 5 months to 3.5 years from 10 software teams that used the Gender Inclusiveness Magnifier (GenderMag) to achieve the gender inclusivity quality attribute. GenderMag is a method for detecting and fixing gender inclusivity issues in software. In this article, we summarize several practices the teams devised and pitfalls they encountered.

What if the software you create excludes diverse populations, marginalizing people who "don't fit," where "not fitting" can simply arise from the user being different from those who created the software? Besides the obvious ethical issues of creating software that is noninclusive, there are economic impacts, such as a loss of market share because fewer customers find the product useful. In response, during recent years, some organizations have begun to see inclusivity across diverse populations as a worthwhile quality attribute for software.

But how can software teams achieve this quality attribute in ways that are not only effective but cost-effective? To find out, we followed 10 real-world teams to collect their practices and pitfalls for fulfilling one variant of the inclusivity quality attribute: gender inclusivity. Gender is a form of diversity for which software inclusivity issues have been particularly salient (e.g., Cunningham et al.,[1] Ford et al.,[2] Gralha et al.,[3] Kelleher,[4] and Shekhar et al.[5]).

One approach to address such issues is GenderMag, an inspection method that uses customized personas and cognitive walk-throughs to help teams find and fix gender "inclusivity bugs" in software.[6,7] It finds such bugs through five research-based "facets" of individuals' cognitive styles regarding problem solving: motivations, computer self-efficacy, attitudes toward risk, information processing approaches, and learning methods. For example, a software team might uncover an inclusivity bug if a feature is easily discoverable by people who have a tinkering learning style as opposed to a process-oriented learning approach.

Inclusivity bugs identified with these facets are cognitive ones. Because the facets capture well-established (statistical) gender differences in how individuals solve problems, cognitive bugs are also gender inclusivity bugs. Note that cognitively diverse behaviors occur not only between one gender and another but within genders. In our field study, the software teams used this method and Abi as a customized persona[6] to elicit requirements for achieving the gender inclusivity quality attribute.

Method

To investigate how teams incorporated GenderMag into their everyday practices, we used action research, which is a longitudinal, collaborative field study methodology in which problems are addressed in situ right away.[8] We worked with 10 software development teams (four university teams and six from industry) during periods ranging from 5 months to 3.5 years. Some teams had previous experience using GenderMag (half of the industry teams) and some were just starting. Some teams mostly worked on their own using the online GenderMag materials, while others asked us to join some of their evaluations of their projects.

We collected multiple types of data, enabling the triangulation of our

Digital Object Identifier 10.1109/MS.2020.3019540
Date of current version: 22 October 2020

0740-7459/20©2020IEEE

findings. At the end of the data collection period, we offered a poststudy interview and debriefing. We used qualitative analysis with triangulation to ensure the rigor of our results. Full details of the study are in Hilderbrand et al.[9]

Practices: Minimizing Costs and Maximizing Benefits

Abstracting—With Discipline

A characteristic of GenderMag is that it is a concrete method. It takes actual inputs [a customized persona, a scenario, and a specific user interface (UI)] and produces material outputs. Despite this concreteness, teams A and C abstracted beyond their UI instance to other instances of the same UI pattern in their product, eliminating the need to evaluate each instance in its own context. For example, team C evaluated a "representative" analytical reporting dashboard and applied the results across all its product's instances of dashboards. Team member TC-3 said:

> *It's not just for one dashboard even though we tackled just one dashboard. It's a good starting point for all our dashboards.*

However, being overly ad hoc about abstracting led to problems. Team C tried using an older version of its UI as a proxy for the new one, but doing so confused the team members and damaged their ability to imagine what Abi would and would not see. Team member TC-15 remarked:

> *In the real environment, there wouldn't be … these other tabs.*

Moreover, the team failed to evaluate the workflow and features that would be available in the new interface. Thus, the practice of "abstracting beyond" paid off only when it was used with discipline (i.e., only for multiple instantiations of a single pattern) and not for merely "similar" systems.

Faulty designs for which teams lacked decision-making power were less likely to be fixed, resulting in wasted time and additional efforts.

Practice: Abstracting Beyond. Abstracting beyond one session's concrete results to entire UI patterns enabled the reuse of findings and fixes.

Potential Pitfall: Evaluating a Proxy. Evaluating a "similar" system can lead to assessing things that are not in the real system, missing features from real system and/or spending extra time during evaluation trying to keep track of differences.

Buy-In and Control

Buy-in problems arose when a team's decision makers weren't part of the sessions, because then the evaluators couldn't actually fix the inclusivity bugs. Trying to later convince those not present to fix the bugs rarely succeeded. Other cases in which follow-up was not possible arose with external software and application programming interfaces (e.g., when a team's design was integrated with third-party software). These situations left certain teams unable to act on some of their evaluation results.

Potential Pitfall: Beyond Our Control. Faulty designs for which teams lacked decision-making power were less likely to be fixed, resulting in wasted time and additional efforts to convince decision makers.

Abi: Talking "Safely" About Inclusivity Bugs

Besides Abi's prowess at revealing inclusivity bugs—team member TW-523 noted, "[Abi] violates a lot of our assumptions around … our tech"—teams also reported that speaking through Abi (or through any of the personas) brought a level of "safety" to critiquing one another's designs. Abi distanced the critiquing team member from the criticism, helping to avoid implications that one team member thought badly of another's work. Team member TN-190 said:

> *We have the [developers] who designed this UI, and it was like, once they were Abi, they could let go of their ego.*

Practice: Speaking Through Abi. Using Abi eased potentially contentious and uncomfortable design discussions by framing critiques from Abi's perspective.

Calculating Bias

At the end of their GenderMag sessions, several teams calculated their bias rates, which gave an overall picture of the teams' results and enabled simple "biasedness" comparisons with their alternative design ideas (see Figure 1 for one team's calculation). Doing these calculations triggered discussions beyond the big picture. Team member TW-523 remarked:

> *I think a lot of the failings were based on the fact that we assume users will explore the system.*

The calculations also motivated thoughtful reflections on the facets and how widely those aspects occurred. Team members even started identifying their own facets. One team member said in the team TN debriefing recording:

> *My personality falls somewhere between Abi and Tim. I'm a read-the-manual kind of person. I'm super risk averse.*

Practice: Calculating Bias. Calculating a product's "bias rate" enabled teams to have big-picture reflections about their populations and the facets they overlooked and to prioritize fixing the inclusivity bugs they found.

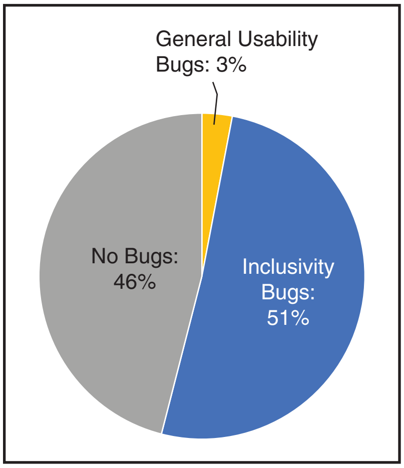

FIGURE 1. One team's bias rate, calculated by dividing the total number of inclusivity bugs it found (i.e., the number of "steps," with a *maybe* or a *no* answer and a facet given as the reason) by the total number of steps evaluated.

Practices Beyond the Session

GenderMag Moments

Team N devised "GenderMag moments"—tiny fragments of a full session triggered just in time by in-the-moment design questions such as, "Should we show these choices alphabetically or in sequence?" Teams familiar with GenderMag simply brought in one step of the GenderMag walk-through questions to answer design questions like these right away.

Team N shared the practice, and seven other teams followed suit. Team A also incorporated GenderMag moments into design meetings to analyze potential fixes' likelihood of addressing the inclusivity bugs it had originally uncovered. Using GenderMag moments in ways such as these reduced the need for full GenderMag sessions, without teams losing the inclusivity analysis benefits of the full method.

Practice: GenderMag Moments. Teams worked out two uses for GenderMag moments:

1. using the GenderMag questions to guide the evaluation of design solutions just in time
2. checking against earlier sessions' filled-out forms to decide whether the fixes would address all the inclusivity bugs they had found.

Real Users' Facets

Four teams harnessed the GenderMag facets for their user studies. Team N was already employing surveys to categorize its user population,

Table 1. Evidence behind each practice/pitfall.

	First GenderMag session	Multiple GenderMag sessions	Follow-up meetings	Interviews	Emails	Evidence in prior literature
Minimizing costs and maximizing benefits						
Abstracting			✓✓			
Beyond control	✓	✓				Burnett et al.[10]
Evaluating proxy	✓✓					
Speak through Abi			✓	✓✓		
Calculating bias	✓✓	✓✓				
Beyond the session						
GenderMag moments			✓	✓✓✓✓✓	✓	
Facet survey			✓	✓	✓✓✓	Vorvoreanu et al.[7]

The checkmarks are instances of the data sources (columns) providing the evidence. For example, we observed evidence of the "facet survey" practice in one follow-up meeting, one interview, three emails, and in prior literature. Excerpted from Hilderbrand et al.[9]

so it devised a facet survey[9] to analyze its users' facet distributions. Teams B and O then used team N's facet questions to analyze their lab studies data. Team O also harnessed its users' facet data to prioritize its inclusivity bug fixes.

Practice: Facet Survey. Bringing facets into survey questions facilitated measuring real users' facet values, which helped teams to do the following:

1. understand their user populations
2. analyze their lab study data
3. measure the effectiveness of their fixes facet by facet.

The Practices Taking Hold

As Table 1 shows, teams found multiple ways of incorporating GenderMag into their existing requirements and design practices. Regarding follow-up, all 10 teams decided to make inclusivity fixes to their products. Of those teams, eight have already applied the fixes, one team had its follow-ups rejected (from the "beyond our control" pitfall), and the last team still has its fixes in progress.

Some teams also reported that GenderMag had impacted their mindsets, making their members more aware of diversity in cognitive styles. Team member TA-PI noted:

> *[This] was not something [we] even were aware of. [We were] not familiar with cog[nitive] styles and how that might affect success when using the product.*

Perhaps the central message behind these teams' experiences is that suspecting your software of gender bias and wanting to fix it is a start, but integrating a systematic process can make the goal actionable. Team member TC-3 said:

> *I thought it was very, very informative. There are some things that we knew we had to change. This … gave us a process.*

Acknowledgements

We thank Claudia Hilderbrand, Christopher Perdriau, Jillian Emard, Zoe Steine-Hanson, and the software teams. This work received funding through National Science Foundation grants 1528061, 1815486, and 1901031.

References

1. S. Cunningham, A. Hinze, and D. Nichols, "Supporting gender-neutral digital library design: A case study using the GenderMag toolkit," in *Proc. Int. Conf. Asian Digital Libraries*, 2016, pp. 45–50. doi: 10.1007/978-3-319-49304-6_6.
2. D. Ford, J. Smith, P. J. Guo, and C. Parnin, "Paradise unplugged: Identifying barriers for female participation on stack overflow," in *Proc. 2016 24th ACM SIGSOFT Int. Symp. Foundations of Software*

ABOUT THE AUTHORS

MARIAM GUIZANI is a Ph.D. student of computer science in the School of Electrical Engineering and Computer Science at Oregon State University. Contact her at guizanim@oregonstate.edu.

LARA LETAW is an inclusive software design researcher and instructor of computer science in the School of Electrical Engineering and Computer Science at Oregon State University. Further information about her can be found at https://eecs.oregonstate.edu/people/Letaw-Larissa. Contact her at letawl@oregonstate.edu.

MARGARET BURNETT is a distinguished professor in the School of Electrical Engineering and Computer Science at Oregon State University. Further information about her can be found at http://web.engr.oregonstate.edu/~burnett/. Contact her at burnett@engr.orst.edu.

ANITA SARMA is an associate professor in the School of Electrical Engineering and Computer Science at Oregon State University. Further information about her can be found at http://web.engr.oregonstate.edu/~sarmaa/. Contact her at anita.sarma@oregonstate.edu.

Engineering*, pp. 846–857. doi: 10.1145/2950290.2950331.

3. C. Gralha, M. Goulao, and J. Araujo, “Analysing gender differences in building social goal models: A quasi-experiment,” in *Proc. 2019 IEEE 27th Int. Requirements Engineering Conf. (RE)*, pp. 165–176. doi: 10.1109/RE.2019.00027.
4. C. Kelleher, “Barriers to programming engagement,” *Adv. Gender Educ.*, vol. 1, no. 1, pp. 5–10, 2009.
5. A. Shekhar and N. Marsden, “Cognitive walkthrough of a learning management system with gendered personas,” in *Proc. 4th Conf. Gender & IT*, 2018, pp. 191–198. doi: 10.1145/3196839.3196869.
6. M. Burnett et al., “GenderMag: A method for evaluating software’s gender inclusiveness,” *Interact. Comput.*, vol. 28, no. 6, pp. 760–787, 2016. doi: 10.1093/iwc/iwv046.
7. M. Vorvoreanu, L. Zhang, Y.-H. Huang, C. Hilderbrand, Z. Steine-Hanson, and M. Burnett, “From gender biases to gender-inclusive design: An empirical investigation,” in *Proc. 2019 CHI Conf. Human Factors Computing Systems*, pp. 1–14. doi: 10.1145/3290605.3300283.
8. G. Hayes, “Knowing by doing: Action research as an approach to HCI,” in *In Ways of Knowing in HCI*, J. Olson and W. Kellogg, Eds. New York: Springer-Verlag, 2014, pp. 49–68.
9. C. Hilderbrand et al., “Engineering gender-inclusivity into software: Ten teams’ tales from the trenches,” 2020. Presented at Proc. 2020 ACM/IEEE Int. Conf. Software Engineering. [Online]. Available: ftp://ftp.cs.orst.edu/pub/burnett/icse20-genderMag-practices.pdf
10. M. Burnett, A. Peters, C. Hill, and N. Elarief, “Finding gender-inclusiveness software issues with GenderMag: A field investigation,” in *Proc. 2016 CHI Conf. Human Factors Computing Systems*, pp. 2586–2598. doi: 10.1145/2858036.2858274.

Rejuvenating Binary Executables • Visual Privacy Protection • Communications Jamming
IEEE SECURITY & PRIVACY
Building Dependability, Reliability, and Trust
SOFTWARE EVERYWHERE
January/February 2016 Vol. 14, No. 1
Policing Privacy • Dynamic Cloud Certification • Security for High-Risk Users
IEEE Symposium on Security and Privacy
March/April 2016 Vol. 14, No. 2
Smart TVs • Code Obfuscation • The Future of Trust
ECONOMICS OF CYBERSECURITY PART 2
May/June 2016 Vol. 14, No. 3
S & P

Digital Object Identifier 10.1109/MS.2020.3027395