1-27-2016

GenderMag: A Method for Evaluating Software's Gender Inclusiveness

Margaret Burnett
Oregon State University

Simone Stump
City University London

Jamie C. Macbeth
Clemson University, jmacbeth@smith.edu

Stephann Makri
City University London

Laura Beckwith
HCI Researcher

See next page for additional authors

Interacting with Computers Advance Access published January 27, 2016

doi:10.1093/iwc/iwv046

GenderMag: A Method for Evaluating Software's Gender Inclusiveness

Margaret Burnett[1,*], Simone Stumpf[2], Jamie Macbeth[3], Stephann Makri[2], Laura Beckwith[4], Irwin Kwan[5], Anicia Peters[1] and William Jernigan[1]

[1]*Oregon State University, Corvallis, OR, USA*
[2]*City University London, London, UK*
[3]*Clemson University, Clemson, SC, USA*
[4]*HCI Researcher, Copenhagen, Denmark*
[5]*The Mathworks, Boston, MA, USA*
[*]*Corresponding author: burnett@eecs.oregonstate.edu*

In recent years, research into gender differences has established that individual differences in how people problem-solve often cluster by gender. Research also shows that these differences have direct implications for software that aims to support users' problem-solving activities, and that much of this software is more supportive of problem-solving processes favored (statistically) more by males than by females. However, there is almost no work considering how software practitioners—such as User Experience (UX) professionals or software developers—can *find* gender-inclusiveness issues like these in their software. To address this gap, we devised the GenderMag method for evaluating problem-solving software from a gender-inclusiveness perspective. The method includes a set of faceted personas that bring five facets of gender difference research to life, and embeds use of the personas into a concrete process through a gender-specialized Cognitive Walkthrough. Our empirical results show that a variety of practitioners who design software—without needing any background in gender research—were able to use the GenderMag method to find gender-inclusiveness issues in problem-solving software. Our results also show that the issues the practitioners found were real and fixable. This work is the first systematic method to find gender-inclusiveness issues in software, so that practitioners can design and produce problem-solving software that is more usable by everyone.

RESEARCH HIGHLIGHTS

- We discuss five facets of prior gender research with ties to males' and females' usage of problem-solving software.
- We present GenderMag, the first systematic method to evaluate gender-inclusiveness issues in problem-solving software.
- We show how GenderMag draws upon and encapsulates these five facets.
- We present three qualitative empirical studies that were used to inform and to validate various aspects of GenderMag, and show the kinds of issues that participants found and how gender of the evaluator interacted with usage of the method.

Keywords: gender; walkthrough evaluations; software inclusiveness

Editorial Board Member: Simone Barbosa

Received 29 July 2015; revised 9 December 2015; accepted 14 December 2015

1. INTRODUCTION

Recent research calls into question the inclusiveness of software that aims to support diverse people in problem-solving situations. The users who tend to be best supported by problem-solving software tend to be those best represented in software development teams (e.g. relatively young,

able-bodied, males), with other users' perspectives often overlooked. Perhaps the best-studied of underrepresented groups' use of software are those with physical disabilities, but even that group remains underserved (Power *et al.*, 2012), and many other underrepresented groups' uses of software remain barely considered (Arjona-Reina *et al.*, 2014; Burnett *et al.*, 2011; Davidson and Jensen, 2013; Joyce *et al.*, 2007; Power *et al.*, 2012).

In the realm of one underrepresented group in software, namely females, Williams recently coined the term 'gender lens' (Williams, 2014), which refers to the need for software development practices that include a gender perspective. In this paper, we present such a lens, in the form of GenderMag (Gender Inclusiveness Magnifier). GenderMag is an inspection method for evaluating problem-solving software from a gender-inclusiveness perspective.

1.1. What is gender?

In this paper, we use the term 'gender' as a socially constructed concept (Butler, 1999; West and Zimmerman, 1987) where gender identification, display and performance might or might not align with biological sex. As West and Zimmerman define it, someone's gender choice affects and is affected by how they manage their 'situated conduct in light of normative conceptions of attitudes and activities appropriate to' the category with which they most identify. We especially emphasize that when someone identifies as a male or female, this is not the same thing as a claim to align with a stereotypical male or female gender role or expression. To reflect this social concept of gender, in this paper, we follow the lead of West and Zimmerman by using the term 'males' as a shorthand for 'people who identify as male', and 'females' to denote 'people who identify as female'.

1.2. Gender inclusiveness

Research over the past decade has emerged showing that the individual differences in how people use software features aimed at supporting problem-solving tend to cluster by gender, and further, that many such features are inadvertently designed around approaches favored more by males than by females. These differences have been found in a variety of problem-solving software; for example, in spreadsheets Beckwith *et al.*, 2005, 2006, 2007; Grigoreanu *et al.*, 2012, in visualizations (Borkin *et al.*, 2013; Tan *et al.*, 2003), in online classwork platforms (Piazza Blog, 2015), in web and home appliance development and scripting (Cao *et al.*, 2010a; Rode, 2008; Rosson *et al.*, 2010) and in intelligent systems (Kulesza *et al.*, 2011; Szafir and Mutlu, 2012).

Further, research shows that designing software to be more gender-inclusive can benefit everyone, regardless of gender. For example, Tan *et al.* (2003) showed that displaying optical flow cues benefited both females and males in virtual world navigation; Grigoreanu *et al.* (2008, 2010) showed how changes to spreadsheet features relating to confidence, feature support and strategy workflows reduced gender gaps while improving everyone's attitudes and feature usage; and Jernigan *et al.* (2015) showed how a tool designed for a diversity of individual styles and situations enabled both female and male novice programmers who did not receive very much in-person help to program as well as novices who received extensive in-person help. These findings are consistent with similar findings in changing educational practices to improve gender inclusiveness. For example, in education, researchers found that pair programming, which was expected to help female computer science students, not only reduced the gender gap but also increased success and reduced attrition among male *and* female students (Berenson *et al.*, 2004; McDowell *et al.*, 2003).

Successful instances like these are encouraging, but mainly what they show is proof of concept, not a path toward inclusiveness. One mechanism to promote inclusiveness that several researchers have advocated is gender-inclusive practices at design time (Bardzell, 2010; Briggs *et al.*, 2014; Williams, 2014). These are important, but they are not a panacea. What is also needed is a *systematic* method that can be used even if few members of the software team are mindful of gender differences, and even if the software is more mature than being in the initial design stages. This points to the following gap: How can ordinary practitioners, with no background in gender research, identify which aspects of *their* software have gender-inclusiveness issues, realize why those issues are issues and thereby know what they should change?

To address this gap, we devised the GenderMag method (Gender Inclusiveness Magnifier). GenderMag evaluates features in problem-solving software from a gender-inclusiveness perspective. At the core of GenderMag are five *facets* of gender differences that have been extensively investigated in the literature. GenderMag encapsulates the facets into *personas* to bring them to life, and embeds the personas and the facets into a *process* based on the Cognitive Walkthrough (CW). The method aims to provide a systematic and practical way for software practitioners (UX professionals, software developers, etc.) with no background in gender research to find gender-inclusiveness issues in the problem-solving software[1] they are producing.

This paper presents the GenderMag method, along with our investigations to inform and evaluate our approach

[1] When we refer to problem-solving software, we mean software features and platforms in which the user is actively trying to work out a solution to some kind of problem or task, such as with the examples at the beginning of this section.

empirically—a formative case study at a company that produces software allowing medical practitioners to customize programmable hearing aids; a formative workshop event in which researchers evaluated Looking Glass (Gross *et al.*, 2012), a tool that teaches middle school students to program 3D animations; and a qualitative laboratory study in which UX practitioners used GenderMag to evaluate Gidget (Lee *et al.*, 2014), a game-like programming environment designed to teach debugging.

2. BACKGROUND AND RELATED WORK

2.1. Gender differences in problem-solving and programming

We have just cited extensive empirical evidence over the past decade showing individual differences that cluster by gender in the ways people use problem-solving software. We now consider these differences at the foundational level.

Five facets of gender differences relating to problem solving that have been repeatedly implicated by research from other fields, such as psychology, education and communications, are:

(i) *Motivation*: Research spanning over a decade has found that females tend (statistically) to be motivated to use technology for what it enables them to accomplish, whereas males' motivations sometimes come from their enjoyment of the technology for its own sake (Burnett *et al.*, 2010, 2011; Cassell, 2002; Hou *et al.*, 2006; Margolis and Fisher, 2003; Simon, 2001). This difference can affect which features of problem-solving software females vs. males choose to use.

(ii) *Information processing styles*: To solve problems, people often need to process new information, and there is extensive research reporting gender differences here too. In essence, when problem-solving, females are more statistically likely to use comprehensive information processing styles—gathering fairly complete information before proceeding—whereas males are more statistically likely to use selective styles—following the first promising information, then potentially backtracking, in 'depth first' order (Cafferata and Tybout, 1989; Coursaris *et al.*, 2008; Meyers-Levy and Loken, 2015; Meyers-Levy and Maheswaran, 1991; Riedl *et al.*, 2010). Each of these styles has particular advantages, but either is at a disadvantage when not supported by the problem-solving software environment. Particularly relevant here are studies tying gender differences in information processing style to software-based tasks, such as with e-commerce web sites (Simon, 2001), software-based auditing (O'Donnell and Johnson, 2001) and working with spreadsheets (Grigoreanu *et al.*, 2012; Subrahmaniyan *et al.*, 2008).

(iii) *Computer self-efficacy*: One specific form of confidence is *self-efficacy*: a person's confidence about succeeding given a specific task (Bandura, 1986). Self-efficacy matters to problem solving because a person's self-efficacy influences their use of cognitive strategies, amount of effort put forth, level of persistence and strategies for coping with obstacles (Bandura, 1986). Empirical data have shown that females tend statistically to have lower computer self-efficacy than males, as one would expect given phenomena like stereotype threat, and non-inclusive work environments and education practices (Appel *et al.*, 2011; Huffman *et al.*, 2013; Luger, 2014). Self-efficacy levels, in turn, affect people's behavior with technology, such as which features they choose to use and how willing they are to persist with hard-to-use features (Burnett *et al.*, 2010, 2011; Durndell and Haag, 2002; Hartzel, 2003; O'Leary-Kelly *et al.*, 2004; Piazza Blog, 2015; Singh *et al.*, 2013). Fortunately, features designed explicitly for diverse self-efficacy levels have been shown to be preferred by everyone (e.g. Grigoreanu *et al.*, 2008).

(iv) *Risk aversion*: Studies have shown that females tend statistically to be more risk-averse than males (Dohmen *et al.*, 2011), surveyed in Weber *et al.* (2002) and meta-analyzed in Charness and Gneezy (2012)—in numerous decision-making domains, such as in ethical decisions, investment decisions, gambling decisions, health/safety decisions, career decisions and others. In contrast, we have been unable to locate any study in any domain reporting males to be more risk-averse than females. Applying these findings on risk aversion to software usage suggests that risk aversion may impact females' decisions as to which feature sets to use.

(v) *Tinkering*: Research across age groups and professions reports females being statistically less likely to playfully experiment ('tinker') with features new to them, compared with males. However, studies also show that when females do tinker, they are more likely to reflect more in the process and thereby sometimes profit from it more than males do, and further, that some males tinker excessively (Beckwith *et al.*, 2006; Burnett *et al.*, 2010; Cao *et al.*, 2010a; Chang *et al.*, 2009; Hou *et al.*, 2006; Rosner and Bean, 2009). One effect of these differences in tinkering behaviors is their impact on which features of software females vs. males will elect to use, especially when a design choice underlying the software product is that users will learn new features by exploring and tinkering with them.

These facets play out in software-based problem-solving situations in a variety of ways, including which features females and males choose to use, the ways they use them and the strategies they employ involving such features. The following examples help us to illustrate this point.

First, consider spreadsheets, a common setting for problem solving about numeric calculations such as for budgets, grades and finances. In a study of Seattle-area experienced Excel users working with Excel (Beckwith *et al.*, 2007), females' self-efficacy predicted their level of success completing a task, but the same did not hold true for the males; for males, self-efficacy did not matter to how successful they were. This translated to feature use for females: low self-efficacy females relied more than males did on the 'familiar' type of features, particularly value edits. At first glance, a possible reason might seem to be that females were simply better judges of their lack of comprehension of the new features, but the evidence does not support that reason: females' comprehension of the software features was no different than the males' and was not predicted by self-efficacy. In fact, this study re-confirmed other studies' findings of self-efficacy playing out differently for females vs. males (e.g. Burnett *et al.*, 2011).

Our second example involves a different kind of problem-solving software: customizing intelligent systems. Intelligent systems, such as email spam filters and recommender systems, learn computational behaviors customized to one end user and these learned behaviors sometimes require adjustment ('debugging'). Here, one facet that turned out to be very relevant was that of information processing. In one study in which end users attempted to guide an intelligent system to better sort emails into folders by pointing out keywords in the email messages (Stumpf *et al.*, 2008), females spent significantly more time than males working with the system, and also produced more thorough results. This was because females used the provided features more comprehensively (as per the information processing facet above), providing the system with significantly more keywords than males did even though they considered the same number of email messages. Another study in this domain (Kulesza *et al.*, 2011) found that females had significantly lower self-efficacy than males, had more difficulties choosing which keywords to select (a 'selection' barrier) and how to proceed with guiding the intelligent system (a 'design' barrier). Females also more often than males encountered these selection barriers in a sequence, repeatedly running into the same barriers (Kulesza *et al.*, 2011).

Web development and scripting provides a third example domain. In a study of web development by end users (Cao *et al.*, 2010a), as with the above studies, females had lower self-efficacy and focused their efforts on familiar webservice features (versus unfamiliar webservice features) significantly more than the males did. Rosson *et al.*'s study of web developers also showed suggestive gender differences in the use of novel web-based database features that are consistent with these findings (Rosson *et al.*, 2007).

Fourth, a multi-study (Burnett *et al.*, 2010) looked at generalizable patterns across a wide range of problem solvers ranging from administrators to professional programmers using a variety of problem-solving software. The multi-study involved a gender-based secondary analysis of almost 3000 participants from multiple studies' data at a large software company, including, two studies of hobbyist programmers using Visual Studio Express, two studies of professional software developers using Visual Studio, as well as technical problem-solving practices of multiple populations using a variety of other platforms. The results showed significant gender differences across all programming environments and populations as to which features males and females elected to use, as to males' and females' tinkering and exploring behaviors and between males' or females' technical problem-solving confidence. Further, as with the other studies reported in this paper, the confidence differences were not the sole explanation for the differences in feature usage and tinkering. Table 1 summarizes the results of the multi-study.

We also mentioned in the Introduction several examples in which the above gender differences were accommodated through more inclusive feature design (Grigoreanu *et al.*, 2008; Tan *et al.*, 2003). Other examples of supporting these differences through more inclusive designs are Storytelling Alice (Kelleher *et al.*, 2007), in which differences in female vs. male motivations to use technology were leveraged to increase middle-school girls' learning of computer programming, and Gidget (Lee *et al.*, 2014), a game designed to teach

Table 1. Results of a five-study analysis of gender differences in feature usage (RQ1), tinkering (RQ2) and computer-related confidence/self-efficacy (RQ3).

	IT-support users: (#1) survey	Hobbyists: (#2) survey, (#3) field interview	Professional developers: (#4) needs survey, (#5) beta-testers survey
RQ1: Which features	√ (#1)	√ (#2)	√ (#4)
Interesting example: Wizards	√ (#1)	√ (#2)	
RQ2: Tinkering, exploring	√ (#1)	√ (#3)	√ (#5)
One aspect: Attitude re: new technology	√ (#1)	√ (#3)	√ (#5)
RQ3: Confidence differences	√ (#1)	√? (#2), √ (#3)	√ (#5)
One aspect: Evidence of ties with only *some* differences	√ (#1)	√ (#2)	√ (#5)

Note: (#study) denotes which study produced each result. √ denotes statistically significant differences, except where (#study) refers to a qualitative analysis. The '√?' refers to nuances in the hobbyists' confidence results (Burnett *et al.*, 2010).

programming in a gender-inclusive way. Gidget's gender inclusiveness comes from innovating certain programming environment characteristics. For example, it portrays the computer as fallible and personifies error messages (Lee and Ko, 2011; Lee *et al.*, 2014). A contributing technology to Gidget is the Idea Garden. The Idea Garden supports diversity in a variety of ways, one of which is presenting explanatory help in ways that are compatible with both females' tendencies toward comprehensive information processing and males' tendencies toward depth-first information processing (Cao *et al.*, 2013; Jernigan *et al.*, 2015). (We will return to Gidget later in this paper).

Given how significantly such gender differences are tied with software usage, how should developers proceed? The GenderMag method aims to enable software developers to answer that question for themselves in the context of the software products they are producing.

2.2. Analytical evaluation and personas

The GenderMag method is an analytical method for evaluating usability. Analytical methods rely on expert analysis, supported by guidelines, principles or prompts. They can be less labor-intensive than user testing and can reveal problems early in the design process, when they are less expensive to fix (Blandford *et al.*, 2008). The Cognitive Walkthrough (CW) is one such analytical evaluation method.

The CW is a particularly good fit to GenderMag's scope of problem-solving software, because the CW was originally developed from theories of problem solving (Anderson, 1987; Greeno and Simon, 1988) and learning by exploration (Polson and Lewis, 1990; Polson *et al.*, 1992). Because the GenderMag method is based in part on the CW, we describe CWs in detail here.

The CW focuses specifically on ease of learning (Blandford *et al.*, 2008; Lewis *et al.*, 1990; Mahatody *et al.*, 2010; Wharton *et al.*, 1994) and supports systematic evaluation of how a first-time user would carry out a task by using interface features. In a CW, a team of evaluators 'walks through' the interface step by step, evaluating the interface's usability and learnability at each step, in the sequence a user would do when completing some particular task for the first time.

The original method consisted of a page with brief questions, and also assumed a background in Cognitive Science (Wharton *et al.*, 1994). Since then, the method has evolved over several iterations (Wharton *et al.*, 1994). The first iteration made it more formal and complex (Lewis *et al.*, 1991), but problems with the usability of the method and the need for Cognitive Science knowledge as a prerequisite still left it difficult to use. A simplified version then emerged (Wharton *et al.*, 1994). This version, which is often cited and applied today, did not require the evaluator to place as much emphasis on understanding the user's explicit and implicit goal structures for particular walkthrough steps. Several extensions to the Wharton *et al.* method have since been developed, with different foci and for different contexts. In 2010, Mahatody *et al.* identified 11 CW variations: Heuristic Walkthrough, Norman Cognitive Walkthrough Method, Streamlined Cognitive Walkthrough, Cognitive Walkthrough for the Web, Groupware Walkthrough, Activity Walkthrough, Interaction Walkthrough, Cognitive Walkthrough with Users, Extended Cognitive Walkthrough, Distributed Cognitive Walkthrough and Enhanced Cognitive Walkthrough (Mahatody *et al.*, 2010). There is also a Programming Walkthrough variant especially for evaluating programming environments (Bell *et al.*, 1991). In developing GenderMag, we drew from the Wharton *et al.* version (Wharton *et al.*, 1994), and from a more recent streamlined version of the CW (Spencer, 2000), which suggests providing preparatory materials to the team in advance and a strong facilitator within the team to keep the team on track and to avoid lengthy design fixes and discussion.

In the Wharton *et al.* CW, evaluators perform a CW in two phases (Wharton *et al.*, 1994). In the Preparatory Phase, they describe the target user, the task for evaluation and an 'ideal' (or at least correct) sequence of goals, subgoals and respective actions to achieve the task. Then, in the Analysis Phase, they use a prototype of the system to systematically work through the 'ideal' subgoal sequence as if they were the target user, using a set of questions (acting as prompts) to structure their evaluation and uncover possible usability or learnability issues. For each subgoal step, evaluators ask whether users will have formed this subgoal as a step to achieving their overall goal. Not doing so may mean that users might not reach their overall goal, or get stuck. For each of the action steps, the evaluators ask three questions: (1) whether users will note that the action is available to them, (2) whether they will associate the intended effect with the action and (3) whether they will understand that they have made progress toward completing the task. Negative answers to these questions indicate the presence of potential issues that might affect usability and learnability.

The CW method has several strengths. Lewis *et al.* (1990) found that the CW method is more robust than Heuristic Evaluation or traditional think-aloud user studies in terms of variability in evaluator performance. It has been suggested this might be due to its structured nature (Hertzum and Ebbe Jacobsen, 1999). Another strength of the CW method is that it can uncover design errors that may impede novices' learning by doing, but it can also uncover usability issues that extend beyond ease of learning (Mahatody *et al.*, 2010; Wharton *et al.*, 1994). This strength has been attributed to its unconstrained nature (Hertzum and Ebbe Jacobsen, 1999) and correlation of ease of learning with ease of use and functionality (Mahatody *et al.*, 2010; Wharton *et al.*, 1994). It can be used early, in the design phase with early stage prototypes, to uncover errors (Spencer, 2000), and can also be used later, throughout design and development phases (Wharton *et al.*, 1994). Another strength is that the method can illuminate what background knowledge the user should possess to complete tasks (Wharton *et al.*, 1994). A CW strength

particularly pertinent for uncovering gender issues is that the CW can reveal assumptions and misconceptions about the user that the designer might have unwittingly built into the system (Mahatody *et al.*, 2010; Wharton *et al.*, 1994).

The CW also has weaknesses (Hertzum and Ebbe Jacobsen, 1999; Mahatody *et al.*, 2010; Wharton *et al.*, 1992, 1994). For example, choices made in task selection and their decomposition into subgoals and actions during the Preparatory Phase have important consequences on finding issues during the Analysis Phase. Tedium can also be an issue: the same questions are asked multiple times and this can become repetitive for an evaluator.

Perhaps the most important weakness from a diversity/inclusiveness perspective is the danger of describing users in very high-level terms (e.g. 'people who use existing ATM machines' (Wharton *et al.*, 1994)), which may encourage anchoring or stereotyping (Hertzum and Ebbe Jacobsen, 1999). This weakness could be particularly detrimental to the GenderMag goals of helping designers make informed decisions about gender differences relevant to software usage.

To head off this weakness, the GenderMag method includes a set of *faceted personas* to describe a target set of female and male users of the software being evaluated, embedding the facets implicated in problem-solving differences described in Section 2.1. A persona is a vivid description of an 'archetype' of some subset of a system's intended users, including their goals, motivations and attitudes (Adlin and Pruitt, 2010; Cooper, 2004), and personas are becoming increasingly popular in UX practice. Research on the usage of personas shows that designers often use personas to communicate about user needs during design phases of software development, such as via ideation and role-playing during informal tests of prototypes (Friess, 2012; Matthews *et al.*, 2012; Nielsen and Storgaard Hansen, 2014) although a few researchers also suggest their use with analytical evaluation methods like the CW (Adlin and Pruitt, 2010; Friess, 2012).

The creation of personas requires care. For validity and credibility, personas need to be based on qualitative and/or quantitative empirical data about target users (Adlin and Pruitt, 2010; Faily and Flechais, 2011; McGinn and Kotamraju, 2008; Pruitt and Grudin, 2003). For applicability and 'buy-in', they also need to be customizable to some extent (Adlin and Pruitt, 2010), but only in aspects that do not interact with the persona's validity. In keeping with these recommendations, we derived our personas from previous qualitative and quantitative gender studies, and explicitly defined which parts are customizable, as we explain further in Section 3.

3. THE GENDERMAG METHOD

3.1. The method

GenderMag is an evaluation method with which software practitioners can evaluate the problem-solving software they design and produce. The method focuses on the five *facets* of gender differences that we described in Section 2.1, encapsulates them into *personas* to bring them to life and embeds use of the facets into a *systematic process* via a gender specialization of the CW (Wharton *et al.*, 1994). More formally:

Definition: The GenderMag method is an analytical method for evaluating software

(i) according to the following *five facets* of gender differences: motivation, information processing style, computer self-efficacy, risk aversion and tinkering;
(ii) which are encapsulated into a set of *faceted personas*, each of which has a *gender* and has *research-based facet values* for all five facets;
(iii) using a *gender-specialized* CW *process* that integrates references to the facets and to the selected persona throughout.

To *instantiate* the GenderMag method to evaluate a particular software product, the evaluation team selects one or more personas from the GenderMag persona set, optionally customizes the selected personas in the customizable portions of the persona and performs the set-up required for CWs (i.e. defining an ideal sequence of each task to be evaluated) in the Preparatory Phase. The evaluation team then uses this instance of GenderMag in the Analysis Phase to evaluate their own software/prototype by following the gender-specialized CW with each persona they have selected. We explain each of these aspects in the next subsections.

To facilitate GenderMag's instantiation, we have created a GenderMag kit, which contains practical instructions on how to prepare for and conduct the GenderMag CW process, the set of personas and examples and forms. The kit is available at http://eusesconsortium.org/gender/.

3.2. The facets and their integration into personas

There are more than five facets that could be obtained from gender theory and empirical literature, but it seems unreasonable to expect GenderMag users (evaluators) to keep a large number of facets in mind throughout an evaluation. Thus, we settled upon five facets as the maximum we would include. Including only five facets required us to accept the limitation that there are important gender-inclusiveness aspects that influence problem solving but would have to be omitted; however, we accepted this trade-off to support the method's usability. As to *which* five facets we should include, we iterated over this choice through our formative studies. Our criteria were that the facets (1) needed to be *extensively researched* in the literature, (2) needed to be *usable* by ordinary software developers or user experience (UX) practitioners who had no prior background in gender research and (3) needed to have implications for software usage. This process ultimately resulted in the list of facets whose provenance we discussed

in Section 2.1: *motivation*, *information processing*, *computer self-efficacy*, *risk aversion* and *tinkering*.

Using empirical data for these five facets, we incrementally began to create four personas as follows:

For each facet, we considered its range of possible values, and how individuals identifying with each gender cluster across those values. To illustrate, Fig. 1 shows one facet by gender, using data from (Burnett *et al.*, 2010).

Each of the four personas in the GenderMag persona set—Tim, Abby, Pat(ricia) and Pat(rick)—has a value for each of the five facets, and background consistent with those facet values. Together, the four personas cover a wide sweep across these facet values:

(i) We assigned to Tim the facet values *most frequently* seen in males, choosing as a tiebreaker those most different from those seen frequently in females. Thus, Tim represents a large fraction of males (as well as a few females), as in 'Value A' from Fig. 1.

(ii) We assigned to Abby the facet values frequently seen in females that are *most different* from those seen in males. Thus, Abby represents a large fraction of females (as well as a few males), as in 'Value C' from Fig. 1. Intuitively, Abby is meant to represent the 'opposite' of Tim in terms of the five facets.

(iii) We assigned (identically) to the two Pats facet values that combined (1) facet values often occurring for females with (2) facet values somewhat less often occurring for females with (3) facet values often occurring with both groups, resulting in a composite along the lines of 'Value B' from Fig. 1. The two Pats are identical except for their genders. One aim of Pat(ricia) is to combat inappropriate stereotyping of females by showing nuanced differences (and likewise for Patrick and males). The identical Pats together also aim to raise awareness that the important differences relevant to inclusiveness lie in the facets themselves, and not in a person's gender identity. That is, they demonstrate that, although individual differences often cluster by gender, the gender label itself is not the point—the road to inclusiveness lies in the facets. By communicating this through Patricia's and Patrick's commonalities, we aim to encourage evaluators to think in terms of the facets ('is this feature effective for people who have a comprehensive information processing style?') as the road toward inclusiveness across genders.

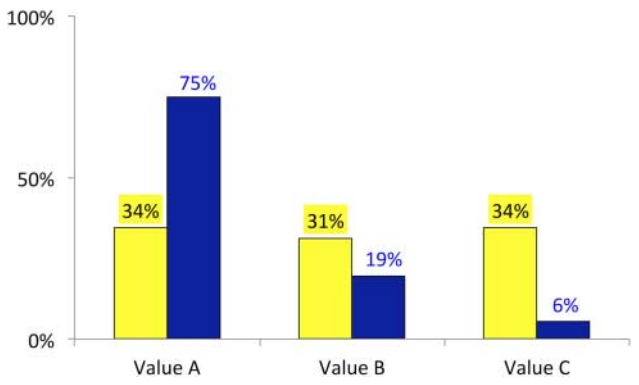


Figure 1. Values for one of the facets. Note that, although females' values (light yellow) are fairly uniformly distributed among Values A, B and C for this facet, the males' values (dark blue) fall much more into Value A than into the other two values. Thus, if Value A is the only one supported at this time in the software, adding support for Value B and Value C would improve inclusiveness for both females and males.

Thus, these four personas are charged with raising awareness of the individual differences that often cluster by gender, and to cover a wide range of the facet values from the literature. For example, Abby's, Patricia's and Patrick's *motivation* to use technology comes from what it can accomplish, whereas Tim enjoys technology for its own sake. As a more nuanced example, Abby prefers ways of learning new features other than *tinkering* (e.g. via tutorials); Tim, Patricia and Patrick all tend to prefer tinkering, but Patricia and Patrick go about tinkering differently than the way Tim does.

Table 2 enumerates all of the personas' similarities and differences for each facet, and all four personas are shown in Figs. A1–A4 in Appendix.

3.3. Personas and stereotyping

Personas, by definition, represent a group of users (Marsden, 2014; Turner and Turner, 2011) with the facet value the persona includes; personas are archetypes of user groups. In our context, this raises a risk of inappropriate gender stereotyping.

We considered several ways to ameliorate this risk. At first glance, it might seem that the answer could lie in somehow removing gender from the personas. However, this is not a promising solution because, with supposedly gender-neutral terms like 'user', most people envision males (Bradley *et al.*, 2015), which would be at odds with our goal of encouraging them to deeply consider males *and* females. This phenomenon is in keeping with Luger's argument that ignoring/removing gender merely hides implicit stereotypical assumptions about gender, making them harder to address (Luger, 2014). Thus, our approach instead goes for explicitness, putting faceted females and males squarely in the center of the evaluation effort, thereby encouraging the feelings of empathy that personas' person-like presentations can generate (Grudin, 2006).

Given use of gendered personas, we have taken three measures to ameliorate the risk of inappropriate stereotyping. We have already alluded to the first two—first, that the four personas show nuanced *within-gender* differences; and second, that the two Pats' identical facet values but different genders aim to particularly emphasize that inclusiveness issues lie not

Table 2. The personas' facet values.

	Purpose	Motivation for using	Information processing style	Computer self-efficacy	Risk aversion	Tinkering
Tim	Tim represents users with facet values most common in males, as in Fig. 1's 'Value A'	Technology is a source of fun	Selective	High (Fig. 1's 'Value A')	Risk-tolerant	Prefers; sometimes tinkers to excess
Abby	Abby represents users whose facet values statistically associated with females are most *dissimilar* to Tim's, as in Fig. 1's 'Value C'. As such, it provides the strongest 'lens' for identifying issues that male development teams may not have noticed before	Wants what the software/ feature can accomplish	Comprehensive	Low. (Fig. 1's 'Value C')	Risk-averse	Avoids
Pat(ricia) Pat(rick)	The Pats (Patricia and Patrick) represent users with values like 'Value B', i.e. statistically associated with many females and some males. Identical female and male personas aim to guard against overgeneralizing	Wants what the software/ feature can accomplish	Comprehensive	Medium. (Fig. 1's 'Value B')	Risk-averse	Prefers; tinkers reflectively

All facet values are based on extensive empirical data. For example, the computer self-efficacy column reflects computer self-efficacy data like Fig. 1, which shows self-efficacy data reported in Burnett *et al.* (2010).

in broad *between-gender* groupings, but in each facet's range of possible values.

Third, the personas explicitly counteract a number of common assumptions not supported by data (Churchill, 2010). One example of such an assumption is with gender and mathematics, an area closely associated with computing. Recent research has shown that when stereotype threat is controlled for, there are no differences in male and female mathematical performance (Else-Quest *et al.*, 2010). Therefore, all four personas are equally proficient with accounting-level mathematics. In fact, all four have equivalent background, job title and responsibilities, math skills, domain knowledge and skill with the technology that they use regularly. All of them even like to play computer games as per research showing that about the same number of males and females play games (ESA, 2015) although the particular games they like sometimes differ (ESA, 2015).

In keeping with these measures, GenderMag constrains personas that have the same job title and responsibilities to be *entirely identical* in everything else too, except empirically established differences. All differences beyond those of the five facets must fulfill these three constraints: (1) they must be empirically supported, (2) they must not suggest a difference in intelligence or education and (3) they must align with that persona's facet values or skill level.

3.4. Persona customizability

Within these bounds, personas must be customizable, so that the software team ultimately using GenderMag can relate to the personas. For example, a product aimed at professional chefs may need a professional chef instead of an accountant representing the user, the Sudoku game may become passé, and a software team in Brazil may not empathize much with a user from Wales. Thus, we have made explicit the parts that can be customized without losing the essence of the four personas (Fig. 2). Of course, if an evaluation team changes one persona's unshaded sections, they must also change all the other personas accordingly.

3.5. Tying the faceted personas to a systematic process

GenderMag connects these personas to a gender-specialized CW at a fine granularity. Our primary specialization is that the gender-specialized CW explicitly encourages reflection on the facets in the personas at every step of the evaluation, to help

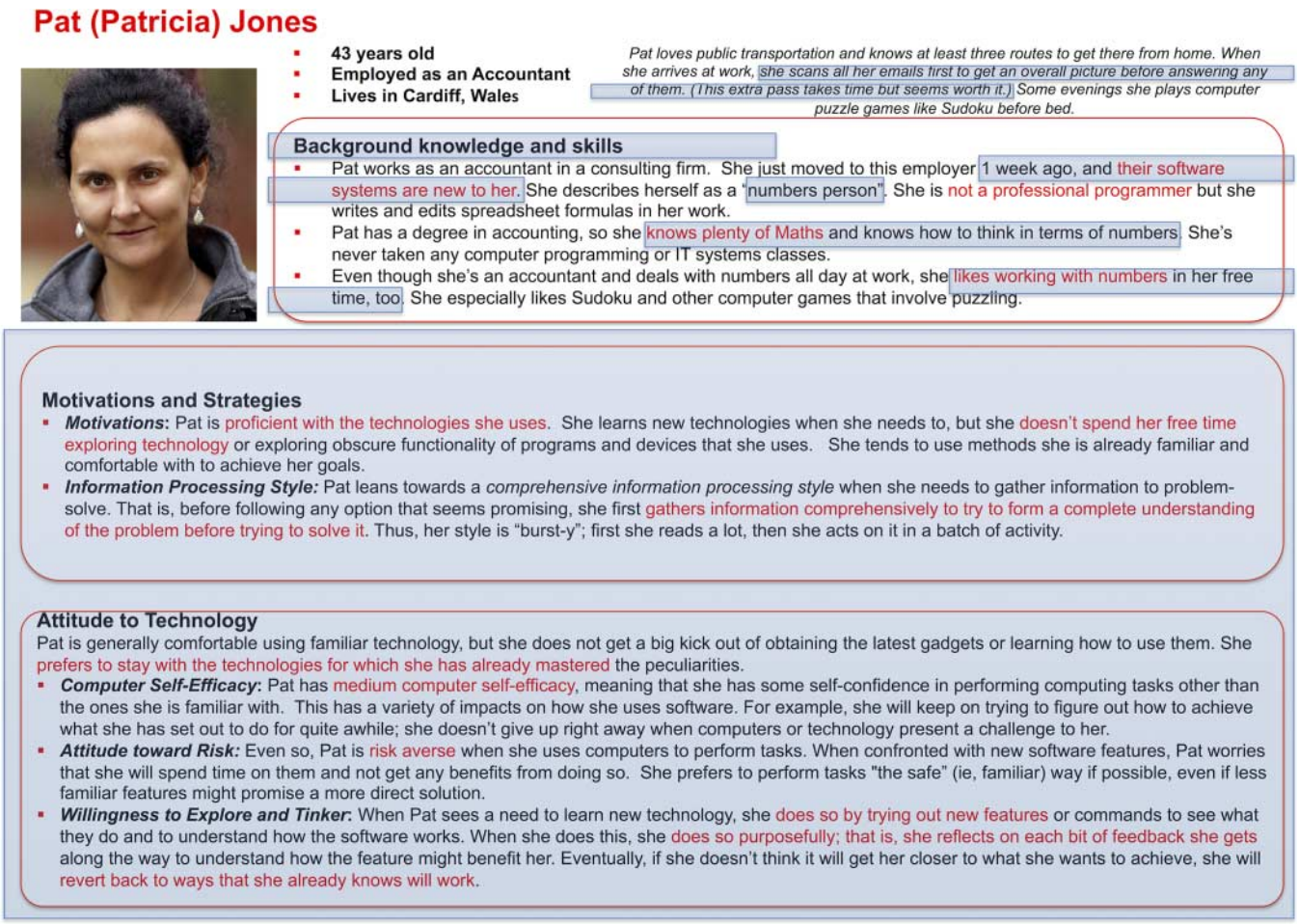


Figure 2. The parts of the personas that an evaluation team is allowed to change, illustrated with Patricia. We have shaded the parts that are not changeable, but the evaluation team can tailor unshaded parts to reflect the target user population. (Appendix shows the unshaded Patricia.)

For each subgoal:

Will <persona name> have formed this sub-goal as a step to their overall goal?
Yes or no: _____
Why? (Especially consider <persona name>'s *Motivations/Strategies*.)

For each action:

1. Will <persona name> even notice that the correct action is available?
   Yes or no: _____
   Why?
2. Will <persona name> associate the correct action with the effect they are trying to achieve?
   Yes or no: _____
   Why? (Especially consider <persona name>'s *Knowledge/Skills, Motivations/Strategies, Self-Efficacy and Tinkering.*)
3. If the correct action is performed will <persona name> see that progress is being made toward a solution to their subgoal?
   Yes or no: _____
   Why? (Especially consider <persona name>'s *Self-Efficacy* and *Attitude toward Risk*.)

Figure 3. The analysis phase of GenderMag CW. The persona name and 'Why' questions are GenderMag's additions over a standard CW.

evaluators remain cognizant of the pertinent evidence-based gender differences throughout the Analysis Phase.

As with the traditional CW, the method has a Preparatory Phase and an Analysis Phase. In the Preparatory Phase, tasks and 'ideal' sequence of actions are defined based on sample forms, just as in the traditional CW. During the Analysis Phase, evaluators walk systematically through pre-defined tasks using a prototype of the system and evaluate whether

the GenderMag persona they are using would have formed the goals, subgoals and 'ideal' action sequences as specified by the developers/designers of the system.

The GenderMag CW includes 'why' questions and explicit references to the current persona's facet values at each goal and action step. In Fig. 3, we show how this changes the traditional, full CW, which is the version we have evaluated so far. (We are now considering moving to a specialization of the more modern, Spencer version instead.)

4. TWO EARLY FORMATIVE STUDIES OF GENDERMAG

We have been iteratively developing the GenderMag method since 2012. As part of its iterative evolution, we describe here two formative studies, which helped to shape the version we evaluated in more detail in a third study (Section 5).

4.1. The method's history and an early case study at company X

GenderMag was first conceived when we received an unexpected email from 'John', a product manager at 'Company X' (Fig. 4).

In essence, the email was a cry for help. It made us realize that—despite all the *research* into gender differences in software usage—there was nothing for *practical use* by industry practitioners without gender research backgrounds, to help them identify their products' gender-inclusiveness issues. In John's case, the gender inclusiveness of his company's software product was critical to its competitiveness.

Toward providing a method useful for such industry practitioners, we began by specializing the CW method. In the instructions, we created to accompany this specialized CW, we provided brief overviews of five facets of gender differences in software use (a slightly different list from that of Section 2.1): Motivation, Risk Averseness, Self-efficacy, Tinkering, Strategy. We also added instructions on how to specify values for each of these facets. For example, our instructions required a score from 1 to 10 to specify the user's level of risk averseness in the user description.

In the summer of 2012, Company X tried out the initial version of the method on their software, and we came and observed them. Our goals were to find out what kinds of difficulties would arise, and whether the method would bring benefits to an evaluation team of non-researchers.

The first difficulty John and his colleagues experienced was in trying to describe the user in the way our instructions required. In wrestling with this problem, John decided to turn to the personas he had seen their Marketing Department use. He knew that personas were often used as a way to describe users, so he proposed to adapt one from the set Marketing had developed. However, he ran into difficulties integrating the five facets we had provided into Marketing's personas, so upon our arrival at Company X, we worked together with John to modify one of the Marketing personas, producing the faceted persona shown in Fig. 5.

To complete the Preparatory Phase, John and his colleagues worked out three task descriptions, which we'll refer to here as Tasks 1, 2 and 3. Task 1 was very basic, intended to help everyone figure out how the specialized CW process worked.

Then, for the Analysis Phase, John gathered four other employees of Company X—an HCI researcher, two software

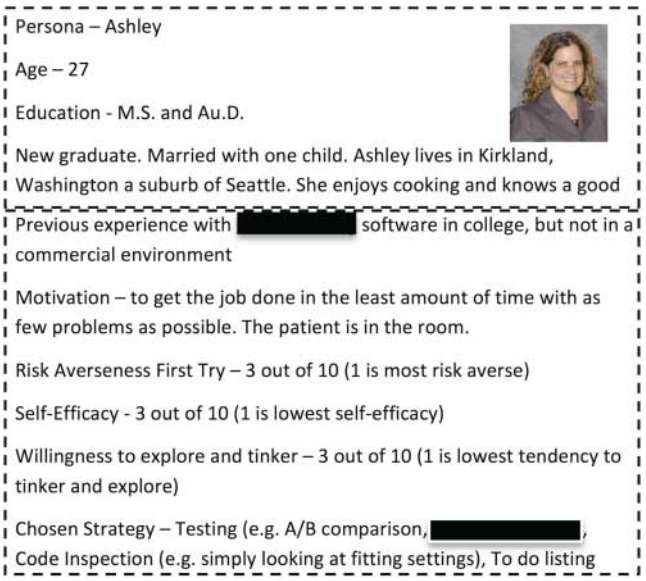


Figure 5. Excerpts from the persona used at Company X. The five facets and their values used in the gender-specialized CW are in the lower part of this persona.

Hello Dr. Burnett,
I am a software product manager for <company name>. I manage software that is used by used by <medical practitioner>s to program <medical device>s for their patients...

85% of <medical practitioner>s are women... I am driving for design ... catering to the female ..., but I am having difficulty defining that in a way that sets parameters and goals for the development team. How do I <accomplish this>?
...
"John"
Product Manager | "Company X"

Figure 4. An unexpected email.

developers and a senior application designer—to help carry out the gender-specialized CW. None had ever done a CW before, and none had carried out any research into gender differences. The group commenced with the CW on Task 1, using the gender-specialized user descriptions, to get up to speed.

Even in the basic Task 1, the group found three issues that could affect the product's gender inclusiveness. This surprised everyone, because Task 1 was so simple, we had thought of it only as a warm-up task. Given these results, after a lunch break, four more company employees (software developers) decided to join in, and together this larger group carried out the evaluation on Tasks 2 and 3.

The after-lunch group revealed a number of additional issues in their software, based on the female persona in Fig. 5. Specifically, in Task 2, a less simplistic task than Task 1, the group found 6 issues, and in Task 3, a fairly complex task, the group found 12 issues. The issues they found tended to be classic usability issues arising from the Gulf of Evaluation and Gulf of Execution (Norman, 2002), such as controls that were not obvious or oddly positioned on the screen, an end user (here, a medical practitioner) having to understand that certain functionalities were available only on certain screens, and lack of feedback as to the effects of an action. Many of the issues they found revealed an underlying expectation by the software design and development team that these things would be clear after a user tinkered and experimented. Indeed, in discussing the issues they found, the team sometimes explicitly gave reasons relating to the two facets of Willingness to Tinker and Self-Efficacy, but this only happened occasionally. Overall, the team thought the method was easy enough to apply during the Analysis Phase without prior experience; this led two of the software developers to ask for more evaluations on other parts of the software.

Company X's in-the-field experience revealed the need for two improvements to the method in order to enable the practitioners to not lose their focus on the facets. First, the team seemed to forget about the facet values fairly often. This emphasized the need to embed reminders to the facets in the CW process itself, so that an evaluation team does not lose sight of them. This led to changes we later made to each question in the gender-specialized CW to finally produce the version presented in Section 3.5. Second, the study revealed the importance of the Preparatory Phase, with faceted personas carefully prepared in advance of the Analysis Phase to concretely capture the facets for evaluators who are not familiar with gender research. We describe next how we incorporated these lessons into the method, and how they played out in our second formative investigation.

4.2. A formative workshop event

Our second formative investigation took place via a workshop event in Fall 2013 at the IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), an academic conference that emphasizes human aspects of programming. Prior to this event, we had made changes to the method, as per the results of the case study at Company X, as follows.

Our first change to the method was to add reminders to the CW of the appropriate facets at each step of the way. Our second change was to eliminate the Strategy facet, which had caused difficulty in the case study; this left the following four: Motivation, Self-efficacy, Risk and Tinkering. The third change was that, instead of using Marketing personas tailored at the last moment to our facets, two HCI researchers drew upon two users from previous inclusiveness research, 'Louanne' and 'F4', and brought them to life by role-playing them (Fig. 6). 'Louanne', a retired businesswoman, represented an experienced, self-taught computer user (not a programmer) who is somewhat accepting of risk and might be willing to tinker (Davidson and Jensen, 2013). 'F4' was a college freshman who was introduced in (Cao *et al.*, 2010b); she was a computer user whose risk aversion, lack of willingness to explore and low self-efficacy figured extensively in her participation in a study of end-user mashup activities (Cao *et al.*, 2010b).

Our task environment was Looking Glass (Gross *et al.*, 2012), a programming environment that builds upon Storytelling Alice (Kelleher *et al.*, 2007). Storytelling Alice is earlier work led by the same researcher as Looking Glass (Kelleher), which teaches middle school students to program

Figure 6. An HCI researcher role-playing 'F4', a college freshman described in (Cao *et al.*, 2010b).

3D animations. In the Preparatory Phase, the Looking Glass researcher provided three tasks; she then served as the expert when questions about the software arose during the Analysis Phase.

For the Analysis Phase, another experienced HCI researcher served as the CW facilitator, and three volunteers at a time from the workshop audience cycled through 10-min stints as the team of evaluators. The CW lasted about 45 min, and went as follows. First, the Looking Glass researcher (Kelleher) projected an instance of Storytelling Alice on the screen and performed the steps of the task. At each step, the facilitator asked the current team of evaluators the CW questions and reminders. After the team gave answers to the questions, the role-players of 'F4' and 'Louanne' each gave answers to the questions from the perspective of their persona. After a few steps, new audience members were rotated into the team to include as many views as possible. Most of the members of this audience-based evaluation team had no prior CW experience.

Applying this version of the method in this way revealed 10 inclusiveness issues[2] that relate to the facets, despite Alice or Looking Glass being a relatively mature platform. The issues identified were mainly associated with three of the personas' facets—Self-efficacy, Risk and Tinkering—such as not exploring system features, sticking to established routines or being surprised by unexpected system actions.

In the workshop event—unlike in the previous case study—the evaluators did not seem to lose sight of the facets, and in fact seemed to become more attuned to nuances of the facets as the CW progressed. This suggests the value of the gender-specialized CW's reminders at keeping the evaluators focused on the facets. We believe this was also partly because the role-players brought the personas to life at every step of the CW.

Of course, HCI researchers who are attuned to gender differences and can role-play the personas are not readily available to UX professionals and software developers, so we decided it was necessary to integrate into the method a set of faceted personas, carefully derived from research data, to make the method more accessible to practitioners. This decision led to the personas we presented in Section 3.

5. A LABORATORY THINK-ALOUD STUDY OF GENDERMAG

Iterating again on the method to incorporate feedback we had gathered from the case study at Company X and the workshop event at VLHCC, we incorporated our first two faceted personas, 'Abby' and 'Tim' (Section 3).

To evaluate and inform the method in much more detail than our previous investigations, we conducted a think-aloud study under the controls possible in a lab setting, using these first two personas. Our aim at this stage of GenderMag's development was to evaluate how professional UX practitioners would apply the GenderMag method. Thus, this study considered the following research questions:

RQ1 (Research Transfer): How do UX practitioners use the facets in the GenderMag method?

RQ2 (Gender): How does a UX practitioner's gender affect method use?

RQ3 (Value): How does the method influence the usability issues UX practitioners find? Are the issues identified of real, practical value?

5.1. Participants and procedures

Ten experienced UX practitioners (4 female, 6 male, 5.25 years mean work experience) took part in the study and, instead of working in a team, evaluated the software individually. All were familiar with CWs, but none had any background in gender research.

Because we were not interested in investigating the gender-specialized CW component alone or the personas component alone, we did not isolate them into separate treatments; rather, we wanted to evaluate the entire method, and to compare its gender-inclusiveness results with an established UX practice. Thus, we randomly assigned participants to one of two conditions: half the participants (referred to as P*i*GM) applied the GenderMag method using the faceted Abby persona and the gender-specialized CW; the other half (referred to as P*i*S) evaluated the software using a standard CW and the Tim persona.

We realized that including Tim with the standard CW could muddy the waters, in that it would remove a clean separation between the GenderMag method (GenderMag group) versus a traditional method (Standard group). Further, using a standard CW does not often include personas—so why include Tim? The answer was fairness: the vague 'the user' the standard CW allows is known to be problematic, sometimes leading to anchoring (Hertzum and Ebbe Jacobsen, 1999). Further, there are proposals to use personas to rectify it (e.g. (Friess, 2012; Adlin and Pruitt, 2010)), so it seemed unfair to arbitrarily choose to ignore this known solution to a known problem. Thus, we decided that the fairest comparison was to provide Tim as a persona of a male 'the user' to the Standard group.

The participants' task was to evaluate a portion of Gidget (Lee *et al.*, 2014), a game-like environment in which novice users program and debug code. Prior to this task, participants filled out a brief background questionnaire, took Gidget's standard short tutorial that explains basic interface elements

[2]Note that these issues were found for personas who do not match the Alice system's intended users. Alice's users are middle-schoolers in an educational setting, whereas our personas represented a much older population with very different motivations, and in entirely different settings. Thus, the issues found should not be viewed as issues about Alice or Looking Glass in its intended usage, but simply as a demonstration of the method's ability to highlight issues *for the personas that are being used during the evaluation*. This points to the criticality of the personas' component of the method—the personas affect the kinds of issues that evaluators will find.

to Gidget users and looked over the persona and CW forms we gave them (GenderMag CW with Abby or Standard CW with Tim, depending on the group). Participants verbalized their thoughts as they worked, and we recorded the session audio and screen activity.

Participants then analyzed three Gidget tasks—Gidget Levels 1, 5 and 20—step-by-step as to the persona's ability to accomplish these tasks, with a maximum of 45 min per level. (One participant (P7GM) went over this time limit and was stopped before completing Level 5.) We selected these levels to represent a range of task difficulty that all Gidget users must overcome to finish the game. Level 1 teaches simple programming constructs. Level 5 has bugs that Gidget users must fix involving arguments and object manipulation. Level 20 introduces functions and is often very challenging to users (Lee *et al.*, 2014)). (Updated versions of these levels can be experienced at www.helpgidget.org.) The facilitator encouraged all participants to refer to the personas, and stepped in only if a participant fell silent or deviated from the method they had been given.

Finally, participants completed an exit questionnaire about their experience. The questionnaire focused in particular on how useful and usable participants regarded the method they had just used, and how likely they were to use it in the future.

5.2. The Gidget environment

Gidget (Lee *et al.*, 2014) (Fig. 7) is an online game designed to teach programming concepts to non-programmers. Gidget is a robot character that tries to save animals endangered by a chemical spill, but its code is faulty. Each level in the game introduces a new animal-saving mission, and in the process introduces new programming concepts in its faulty (Python-like) code. To progress to each new level, users have to solve the problem of how to save the animals in the current level by debugging the provided code.

For this study, we used the Gidget version from 28 May 2014. In this version, the basic UI elements are as follows: Gidget users have three execution buttons ('one step', 'one line' and 'to end') that run Gidget's code and incrementally show the effects of the code on the 'world'. Users can edit the code, can reset all edits made by pressing the 'restore to original code' button, and can inspect objects' properties by clicking on the object in the world.

Note that the participants were evaluating the *entire* environment's ability to enable users to succeed at their problem-solving tasks—namely learning the aspect of programming targeted in each of Levels 1, 5 and 20—not just the UI widgets. For example, for the Level 5 task, an excerpt from the 'ideal' task sequence we provided was as follows:

> subgoal g6: Identify problem: Gidget can't grab the goop because it's too far away…
>
> action g6-a: User runs code to end. [Gidget stops with error].
>
> subgoal g7: Fix problem: Move Gidget to goop location.
>
> action g7-a: User stops code.
>
> action g7-b: User edits code from 'down' to 'down 3.
>
> action g7-c: User inserts code 'left 3' at next line.

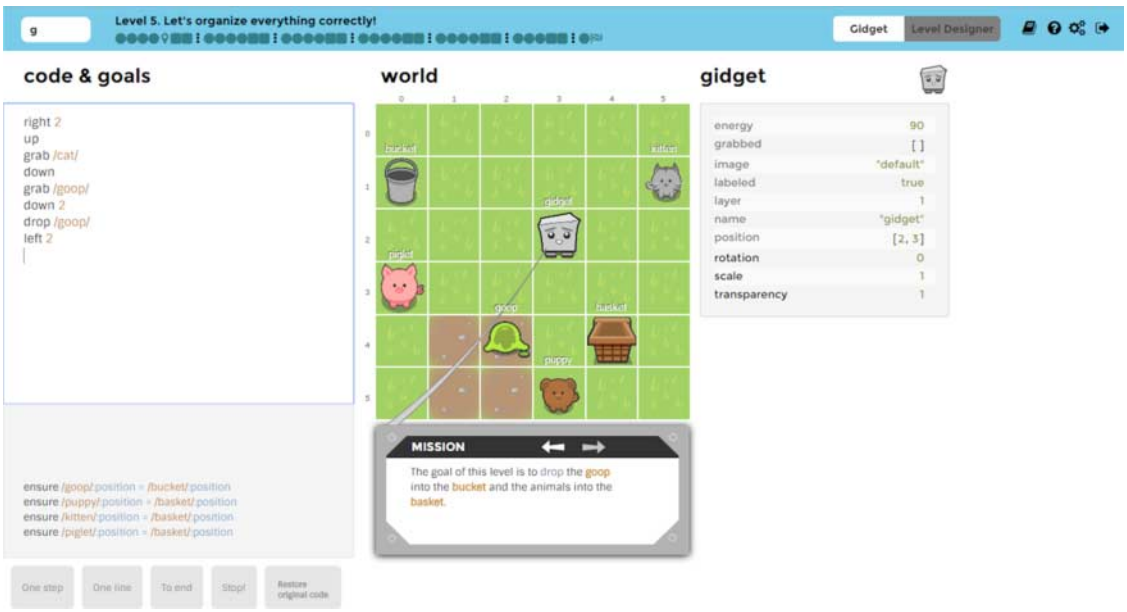


Figure 7. A portion of the Gidget 'game' environment at Level 5, expanded to show Gidget code (left) and the Gidget character in the 'world' (middle). In Level 5, Gidget users debug the code to manipulate several objects.

5.3. Analysis procedures

We transcribed the recordings of the sessions and combined these transcripts with the notes participants made on the CW forms as the basis for our analysis.

5.3.1. Facets referenced

For RQ1 (Research Transfer), a measure of how much of the applicable gender research the GenderMag method transferred to practitioners is in how they used the facets. Thus, for each subgoal and action, we coded the CW forms and the transcripts for each facet that participants referred to (i.e. Motivation, Information Processing, Self-Efficacy, Risk and Tinkering). Two coders independently coded 23% of the data using these codes to check for reliability, with 85% agreement (Jaccard measure of agreement) indicating high reliability of code application. Given this consistency between coders, one of the coders then finished up the coding.

5.3.2. Issue types found

RQ3 considers how the method impacted the usability issues participants found. To investigate this question, we coded the types of issues the participants identified. For this, we analyzed each instance in which participants either said that the persona would struggle or explicitly indicated a problem on the CW forms. We coded the instances into 'types' that reuse and extend those used in previous Gidget studies with real users (Lee *et al.*, 2014). Table 3 shows the provenance of each code.

The rightmost column of Table 3 shows two broad categories of types from the prior Gidget studies: programming concepts and problem-solving anti-patterns (Lee *et al.*, 2014). In the prior studies, users had difficulties understanding certain programming concepts, such as string equality, function calls vs. function definitions, so we looked for all those issues in this study's results as well. We also looked for all the problem-solving 'anti-patterns' reported in those studies (problem-solving strategies that do not lead in a productive direction). We also looked for the 'algorithm barriers' reported in those studies, but did not find them in our data. We then added to that list issue types in our data that had not been reported for the previous Gidget studies.

We evaluated the reliability of this coding scheme on 21% of the data, reaching agreement of 85% (Jaccard measure of agreement) between two coders different than the coders of Section 5.3.1. Then, given this level of consistency, one of these two coders finished up the coding.

6. LAB STUDY RESULTS

6.1. RQ1 (research transfer): from gender research to practical facet usage

Our first research question (RQ1) considers the 'research transfer' question—enabling practitioners to *apply* findings of past gender research to their own situation. Because these findings are encapsulated in the GenderMag facets, we

Table 3. The issue types code set.

Issue type	Description	Provenance
Reinvent the Wheel	Persona deletes code without even reading it	Anti-patterns (Lee *et al.*, 2014)
I don't want to try it	Persona is afraid to try an idea	Anti-patterns (Lee *et al.*, 2014)
When all you have is a hammer	Persona sticks to a programming construct that no longer works	Anti-patterns (Lee *et al.*, 2014)
All-knowing computer	Persona assumes provided code must be ok. Does not read code	Anti-patterns (Lee *et al.*, 2014)
Functions	Persona has problems using functions (e.g. function definitions and calls)	Programming concepts (Lee *et al.*, 2014)
Assertions	Persona has problems using assertions (goal statements)	Programming concepts (Lee *et al.*, 2014)
String equality	Persona has problems using string equality (e.g. case-sensitivity and exact string matching)	Programming concepts (Lee *et al.*, 2014)
Objects	Persona has problems using objects (e.g. objects can be inspected or have behavior)	Programming concepts (Lee *et al.*, 2014)
Dive In	Persona omits testing, just modifies code	Derived from this study's data
Fault localization	Persona cannot find where error is or where a fix needs to be made	Derived from this study's data
What to select	Persona does not know which feature to use (but does know what to do)	Derived from this study's data; similar to selection learning barrier in Ko *et al.* (2004)
Restore	Persona starts over (discard all work) prematurely	Derived from this study's data
Fixed?	Persona does not know if fix is correct	Derived from this study's data
Tracing	Persona has trouble finding code' effects in the 'world'	Derived from this study's data

investigate this question by considering whether and how the UX practitioners made use of the facets.

Although we expected GenderMag participants' facet usage to be higher than those of the Standard participants, the differences between the two groups exceeded our expectations. Four of the five GenderMag participants talked about the facets more than *any* of the Standard participants did, and overall they referred to facets nearly twice as often as Standard participants (Fig. 8, left). Further, this pattern held across every facet: four of the five GenderMag participants referred to *every individual facet* more than any Standard participant did (Fig. 8, right). Since the Standard participants had a faceted persona (but not the GenderMag CW), this suggests that the *combination* of the GenderMag CW process and the faceted persona mattered to participants' application of the facets. Specifically, neither providing personas without providing reminders throughout the process (as in the case study at Company X in Section 4.1), nor providing faceted personas without the GenderMag CW as in the Standard group, seemed as effective at encouraging evaluators' usage of the facets compared with using the entirety of the GenderMag method as a tightly coupled whole.

As to *how* participants used the facets, it was usually by talking about how that facet appeared in their specific persona. In fact, their verbalizations mirrored the gender difference literature very well. This was true of both groups, although as Fig. 8 just showed, more often by the GenderMag group than the Standard group. For example, gender differences in information processing styles were almost paraphrased in these participants' uses of that facet:

> P3GM (*Abby's Information Processing*): 'She's **gathering everything** to understand the problem before trying to solve it.
>
> versus
>
> P4S (*Tim's Information Processing*): 'He just sorta **picks one and tries it out.**

Tinkering was the facet mentioned most frequently, and also illustrates this phenomenon well:

> P2GM (*Abby's Tinkering*): 'Probably not, because . . . she's **not someone who would try**. . .
> P3GM (*Abby's Tinkering*): '. . . and she does bring herself to **tinker**. . .
>
> versus
>
> P5S (*Tim's Tinkering*): 'But, I think Tim is someone that's quite **confident to click around**, so he would find it. . .
> P4S (*Tim's Tinkering*): '. . . He's more of an explorer and a **tinkerer.**

However, the groups' 'how-ness' diverged in interesting ways with the Risk facet. First, the Standard participants rarely brought up Risk at all and second, even when they did, they mentioned only *situational* risks. For example,

> P1S (*Tim's Risk*): 'If I started typing here . . . I click on 'to end,' I assume . . . would be taking me to the end of the line, because that are the terms that I use on my general keyboard. So **there is a danger.**

In contrast, GenderMag participants frequently brought up Risk, and, in most of these cases, they mentioned *personal* feelings of risk that Abby herself would experience:

> P2GM (*Abby's Risk*): '. . . but it does let **her worry a little bit** because this time is different from last time. . .
> P3GM (*Abby's Risk*): ' **She's a bit risk-averse**, so maybe she might not go straight to the end.

This may suggest that either the Abby persona inspired more empathy among the participants than Tim did, and/or that the entirety of the GenderMag method helped promote empathy better. We speculate that both factors contribute, because of the attention the GenderMag participants gave not only to Abby, but also to Abby's facets individually.

Relevant to empathy but not quite the same are these two quotes from male GenderMag participants, who explicitly expressed Abby's value for them in taking on someone else's perspective:

> P7GM (questionnaire): '. . . it was really useful. I mean it's particularly in terms of evaluating from someone else's perspective because it actually forced me to be more objective.

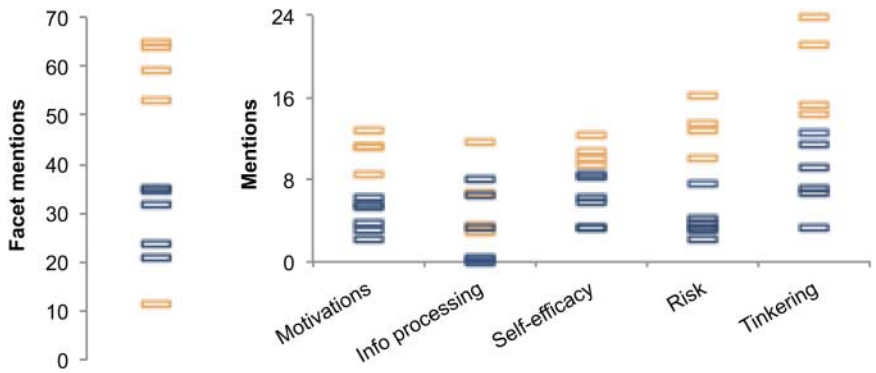


Figure 8. Left: Each participant's total mentions of facets, for GenderMag (light orange) vs. Standard (dark blue). Participants discussed gender facets for GenderMag almost twice as often as they did for Standard. Right: Each participant's mention for each facet, for GenderMag vs. Standard. Except for Information Processing, participants in GenderMag discussed each facet more than in Standard.

> P2GM (questionnaire): 'With a persona, I was able to take user's view further, as in putting myself in the user's shoes – be more aware the walkthrough isn't about me.

In some ways, their comments could be about empathy, but the comments also sound impersonal, suggesting that at least these two males were keeping Abby at arm's length. This brings up the possibility that the evaluators' gender may factor into the process, which we consider next.

6.2. RQ2 (gender): the impact of evaluators' genders

Several hypotheses are possible regarding the genders of the evaluators. One such hypothesis is that if a team is mostly male, they may not benefit very much from GenderMag because they do not relate well to the female personas and/or the female personas' facet values, and are similar enough to the male personas that they already had the intuitions they needed for considering male users' needs. A contrasting hypothesis is that male practitioners and female practitioners might not differ much in their use of GenderMag since the facet connections it provides are integrated into the process in a fine-grained way. Still other possibilities include females being especially engaged with GenderMag due to higher empathy, or conversely, distancing themselves due to a resistance to seeing themselves as being characterized by one of the female personas.

To investigate such possibilities, we analyzed facet usage by gender identity of the evaluator. Figure 9 shows the individual participants' usage of the facets *by gender within groups* (GenderMag on the left, Standard on the right). The female GenderMag participants mentioned Tinkering a great deal more than males did, but otherwise, gender differences were not apparent, suggesting that the participants in general, regardless of gender, were more facet-focused using GenderMag than using Standard.

Considering facet mentions *by group within gender* (Fig. 10: females on the left, males on the right) likewise reveals that *both* female and male UX professionals tended to consider the facets more using GenderMag than their counterparts who used the Standard CW. But it also shows that the difference was especially notable for the female participants—for four of the five facets, GenderMag female participants referred to Abby's facets at least twice as much as Standard females referred to Tim's facets.

The males' and females' comments regarding their empathy toward the personas may help to explain the trends in Figs. 9 and 10. For example, one female GenderMag participant, P2GM, identified strongly with Abby and stated that she and Abby would explore the interface in a similar way:

> P2GM (*female*): 'Although, if she knows **the same as I did, so she got introduced to the same thing I did** … she probably will have worked out what this 'restore original code' means.

Male GenderMag participants also occasionally identified with Abby, but, in general, provided far less evidence of identifying with the Abby persona. At one point, participant P7GM (male) even stated:

> P7GM (*male*): 'I think the **persona's almost immaterial** at this point.

Although the same participant distanced himself from Abby, he nonetheless reported that GenderMag made him objective and unbiased:

> P7GM (*male, questionnaire*): 'I think it actually clarified for me … a consistent lesson of evaluating sort of anything, which is **being objective and not applying your personal bias.**

Thus, overall the results paint a nuanced picture of the impact of the evaluator's gender on use of the method. GenderMag seemed to inspire more empathy with Abby in female participants, and seemed to particularly encourage them to consider Tinkering as they carried out the process. However, although males did not seem to relate as strongly to the Abby persona as the females did, GenderMag still appeared to help male participants consider the persona's facets more

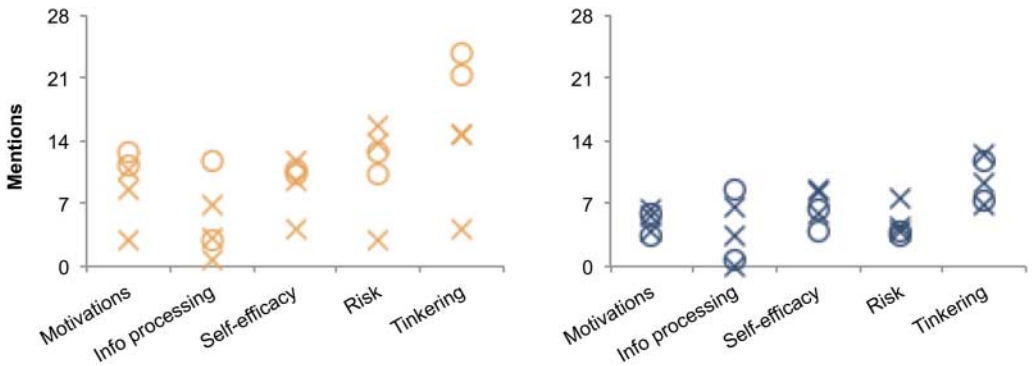


Figure 9. Each participant's mentions of facets, for GenderMag (left) and Standard (right). Circles are female participants and crosses are male participants. GenderMag females had a great deal more to say than males did about Tinkering. Otherwise, little difference was apparent in male vs. female participants' use of facets.

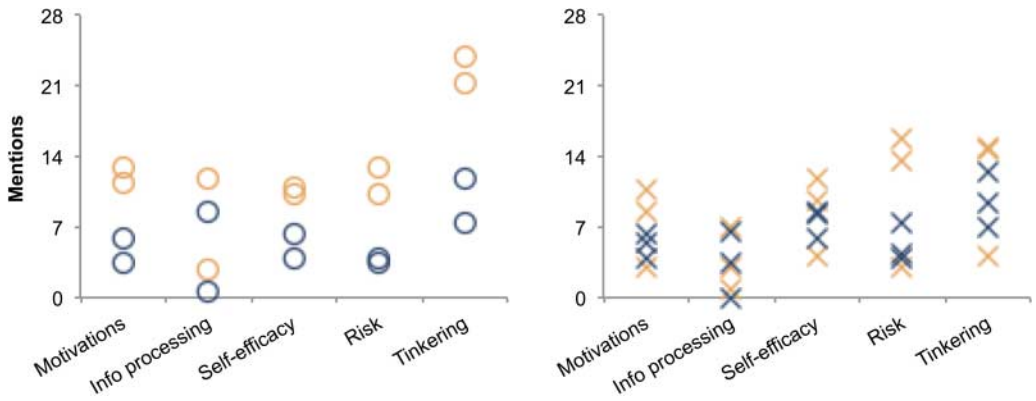


Figure 10. Participants' mentions of facets, for GenderMag (light orange) and Standard (dark blue). Left: Female GenderMag participants mentioned four of the five facets more than Standard females. Right: Male GenderMag participants tended to refer to facets more than Standard males.

frequently than their counterparts who used the Standard CW. These results suggest that using GenderMag as a gender lens had utility regardless of the gender makeup of the evaluation team, but that females might have experienced a greater 'magnification' of gender inclusiveness issues than males did.

6.3. RQ3 (value): the usability issues found

6.3.1. What kinds of issues did participants find, and how?

A key criterion for the effectiveness of any analytical method is its ability to identify usability issues. Numerically, GenderMag participants identified about the same number of issues as Standard participants. We interpret this result as evidence of both the GenderMag method and of the Standard CW (with persona) being effective usability methods from a 'find usability issues' perspective.

However, the ways participants went about identifying issues seemed to differ by group. As Fig. 11 shows, the GenderMag group was more likely to report issues as they related to one of the facet values. Specifically, the rightmost bars of Fig. 11 show that, in contrast to GenderMag participants' issues, almost half of the issue identifications by the Standard participants did not take the facets into account at all.

For example, participants P2GM, P3GM and P7GM all used Risk to find usability issues at action #g2-a, where the user is supposed to stop Gidget at the location of a bug in order to edit the code and fix the bug. In their discussion of this action, participants stated that Abby would press 'restore original code' instead, in part because of her aversion to risk. P7GM put it as follows:

> P7GM (*Risk*): 'That seems a little bit doubtful again, given her persona, to just jump in and start editing. I'd kind of imagine **she would be a bit more cautious** about doing that… She might be inclined to actually click on "restore original code"

The issues the GenderMag group identified for Abby were also often different from those the Standard group identified for

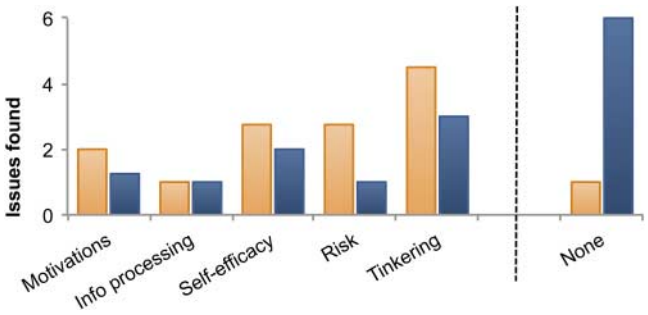


Figure 11. Median number of issues that participants identified using each facet; by GenderMag participants (light orange) and by Standard participants (dark blue). GenderMag participants found far more issues using the Risk and Tinkering facets than Standard participants did; in fact, many issues found by the Standard group did not consider any gender facet at all (rightmost bars).

Tim. To investigate what kinds of issues arose, we categorized the issues as per the code set presented earlier (Table 3). Table 4 enumerates the results. For example, participants identified several issue types (*I don't want to try it*, *When all you have is a hammer*, *Assertions*) as problematic for Abby more often than for Tim, whereas they identified other issue types (*Tracing*, *Reinvent*, *Dive In*) to be problematic more often for Tim than for Abby.

These differences in issue types raise two more questions: whether the issues are 'real', and if so, whether they really align by gender in the same way as they aligned by Abby and Tim. We consider these questions next.

6.3.2. The identified issues' validity

To investigate whether and to what extent the issues the lab study participants found correspond to real issues for real Gidget users, we validated them in three ways. First, we compared the issues our participants identified with a previously published analysis by the development team of

Table 4. The 14 issue types identified by the lab study's participants in order of frequency for Abby, with the percentage of all issues found for the Abby or Tim persona.

Issue type	Abby (%)	Tim (%)	Issue validated?	Gender distribution validated?	How validated?	Follow-up to fix the issue?
I don't want to try it	24	12	√	√	Development team	Hoped for (Outside the development team's responsibility area.)
Fault localization	16	16	√	√	Development team	Yes, dealt with prior to our contact with the development team. Fixes applied: (1) Syntax highlighter had a bug, fixed (2) While typing, syntax and undefined variables are now underlined in red (3) Idea Garden help system: improved guidance to help find the error
What to select	14	9	√	×	Development team	Yes, prior to our contact with the development team. Fixes applied: Improved the built-in tutorial
When all you have is a hammer	11	5	√	√	Development team	Hoped for (Outside the development team's responsibility area.)
Restore	8	0	√	×	Development team	
Functions	8	11	√	√	Development team and Lee *et al.* (2014)	Yes, prior to our contact with the development team. Fixes applied: Idea Garden (Cao *et al.*, 2013) provided help about the concept
Assertions	5	0	√	√	Development team and Lee *et al.* (2014)	Yes, prior to our contact with the development team. Fixes applied: Idea Garden help system provided help about the concept
String equality	3	5	√	√	Development team	Yes, prior to our contact with the development team. Fixes applied: improved Gidget's instructions at the beginning of the level
All-knowing computer	3	2	√	√	Development team and Lee *et al.* (2014)	Yes, prior to our contact with the development team. Fixes applied: Idea Garden help system identified some errors in starting code
Objects	3	2	√	×	Development team and Lee *et al.* (2014)	Yes, prior to our contact with the development team. Fixes: Idea Garden help system provided help about the concept
Fixed?	3	4	×	N/A		N/A
Tracing	3	7	√	√	Development team	
Reinvent	0	4	√	√	Development team	Yes, prior to our contact with the development team. Fixes applied: added a button that says 'clear everything'. The first time they click it, it asks if they are sure because they will lose the hints that the code contains
Dive In	0	25	√	√	Development team	

Gidget's development team validated 13/14 of these issue types. 49% of the validated issues (in the six shaded rows) had gender differences.

problems with using Gidget (Lee *et al.*, 2014). Second, we verified each issue with the Gidget development team by showing the Gidget team the transcript so that they could see for themselves the situations in which our participants reported the issues, and then asked the team whether they had observed these issues in their own users. Finally, we asked the team whether they had noticed gender differences for these issues.

Table 4's 'Issue validated?' and 'How validated?' columns show the results. The development team's observations over the previous 2 years of seeing how males and females used Gidget in camps and lab experiments served as the ground truth for all 14 of the issue types. In total, the Gidget team verified that they had independently observed 13 of the 14 of the issue types. Four of the issue types had formally been published the year before (Lee *et al.*, 2014)—but note that the study participants who found these issues had not seen Gidget or the publication. The verified issue types accounted for 97% of the issue instances that the study participants identified for Abby and 96% of those identified for Tim.

Regarding gender, 10 of the 13 validated issues matched the Gidget team's experience of gender distribution (Table 4, 'gender distribution validated?' column). Of these 10, four were reported and validated about equally across gender, and six (shaded in the table) were reported and validated to differ across gender. In total, these verifications of the gender distribution of issue types covered 81% of the issues the study participants had identified.

Finally, the Gidget team verified that (at least) six of the issue types were important—so important that they need to be fixed. We know this because, at the time of the interview, they had *already* proceeded to fix four of them. (They could not fix the other two as they were outside that team's responsibility.) Perhaps due in part to the issues the Gidget team had fixed in time for the Gidget public release, their software is quite popular with females. At the time of writing, 47% of the users of Gidget's registered users identify as female.

7. DISCUSSION: THE ROAD AHEAD

The studies described in this paper provide proof-of-concept evidence that the GenderMag method can (and did) alert ordinary software practitioners—such as the product manager and software developers of Company X and the UX practitioners of the lab study—to inclusiveness issues in problem-solving software that can impact different genders differently. We have also very recently completed a field study involving industrial uses of GenderMag (Burnett *et al.*, 2016). In that study, four teams of software practitioners (mainly software managers and software developers) at a government agency and at two large hardware/software companies conducted GenderMag evaluations on their own software. All four teams found gender-inclusiveness issues in their own software using the method. These results are encouraging evidence of GenderMag's effectiveness.

Our next research goals revolve around conducting long-term studies in real-world settings to investigate possible obstacles to adopting GenderMag. In fact, in some organizations, there may be barriers to even *trying*, let alone adopting, the GenderMag method. One of our next research goals is to catalog these barriers and to understand the potential changes we might make to the method to address them.

Our previous research has suggested several possible obstacles to adoption that GenderMag could face. First, we have seen instances of *philosophical* obstacles to investigating gender differences. Some people call into question whether females and males behave differently with software at all; our work takes the opposite stance, resting upon the evidence presented in Section 2. As to people who do acknowledge the existence of gender differences, they hold numerous views of those differences. One spectrum of these views runs from essentialist perspectives, which hold that cognitive and behavioral differences between males and females are innate, to social-construct perspectives, which see gender differences as arising through society's attitudes toward gender roles. Although neither end of this spectrum questions whether gender differences exist, these perspectives suggest different directions as to how to address gender differences. That is, if gender is viewed as a social construct, this suggests that a way to address gender differences is by breaking down barriers that may have come about through learned gender roles (e.g. as with stereotype threat (Appel *et al.*, 2011; Zhang *et al.*, 2013)). This is the direction the GenderMag method takes.

In addition to the above philosophical obstacles, certain *organizational* and *practical* obstacles to adoption can arise. For example, some organizations may believe that females are not an important customer group to target, and thus their software does not need to be gender-inclusive. We have also encountered some software teams who were uncomfortable with or unable to investigate gender issues because of their organization's privacy or equal opportunities policies. In other instances, some teams may believe that they do not need GenderMag if the team includes a number of females on it. Indeed, one of our industrial contacts expressed a lack of interest for precisely this reason. This question remains open: data from our lab study suggest that the female participants got a bit *more* out of GenderMag than the males (recall Section 2). Still, regardless of whether it helps to have females on the team, it makes sense for *any* team to use GenderMag if it feels the need; the all-male 'Company X' software development team offers a case in point (recall Fig. 4).

Finally, obstacles can be *methodological*. For example, methodological objections could stem from a team's resistance to one of the GenderMag components, such as use of personas. Indeed, Adlin and Pruitt stress the need for a number of steps to encourage adoption of persona-based methods into organizations, without which persona adoption can fail (Adlin

and Pruitt, 2010). As another example, teams not accustomed to using analytical methods might be pessimistic about the expected cost/benefit of using analytical methods (such as GenderMag) vs. empirical methods. In our lab study, when we asked participants if they would use our method in the future, one participant said precisely this, explaining that it was not viable because it was too '*time consuming*' (Participant P5S). However, within the GenderMag group, all five participants stated they would be likely to use GenderMag in the future; for example, '*for breaking down difficult or complex evaluations into component parts*' (P7GM), to provide '*interesting insights*' (P2GM) and its '*efficiency*' (P6GM). Especially appreciated was GenderMag's potential to spot problems and inform design, for example: '*I could see issues I wanted to fix by redesigning right away*' (P6GM). Still, analytical methods can be as time-intensive as some kinds of 'quick' user studies in industry, and the perceived benefits of user studies might be higher. In fact, one of our industrial contacts, an industry-based UX professional, commented that quick studies involving users seem more persuasive to their software teams and managers than analytical results.

Some of these obstacles to adoption could suggest improvements to the method that can address the obstacles. For example, the philosophical category has already inspired the mechanisms explained in Section 3.3 to guard against inappropriate stereotyping. Others, such as a methodological distrust of persona-based methods, may simply suggest introduction and presentation processes to allay concerns about the method that are unwarranted. Still others, such as organizational climate, may identify organizations and situations that are not right for use of the GenderMag method.

8. CONCLUSION

In this paper, we have introduced GenderMag—the first *systematic* evaluation method for *practitioners* to find *gender-inclusiveness* issues in problem-solving software. At the GenderMag method's core are five facets drawn from an extensive body of research literature on gender differences that can impact use and usability of problem-solving software. The five facets are the central point of the method. That is, by promoting support for the five facets, GenderMag is not ultimately about labeling people by gender, it is about designing for a diversity of individuals' problem-solving facets that happen to cluster statistically by gender.

We have iteratively evolved and empirically informed GenderMag across a range of settings and with a variety of evaluator types. In these investigations, evaluators were software developers, software managers, HCI researchers and UX practitioners; personas represented a medical practitioner, a college student, a retired businesswoman and an accountant; and software products evaluated were a system for customizing medical devices, an end-user programming environment for storytelling and a debugging game. In addition, HCI students and local software developers have used it informally (beta-tested it) to evaluate a programming tool for biocomputing researchers, a support system for travelers, a mobile-based document system and a decision support system to help chemists or environmental engineers choose which materials to use in their manufacturing processes. Emerging results suggest that the scope of GenderMag might be slightly larger than for software that directly targets problem-solving; it seems to be useful in evaluating any interface that is itself complex enough to involve problem-solving (e.g. 'how do I make the system do what I want?'), even if the task being supported by that complex interface is not a problem-solving task.

In all of these uses of GenderMag, the evaluators have *always* found issues. Further, most of the issues they have found were *real* issues, as with the evaluation of the lab study results with the Gidget team. Finally, as the lab study in this paper also illustrated, GenderMag enabled participants to identify gender-inclusiveness issues—even though none of them had a background in gender research.

The method is available in 'kit' form at http://eusesconsortium.org/gender/, and is being beta-tested in several HCI education and production software settings in Denmark, Germany, Singapore, Sweden, the UK and the USA. We aim for its usage to not only continue to inform the method itself, but also to inform the expansion of its personas corpus and our understanding of the boundaries of the method's scope.

Ultimately, this research aims to help software teams avoid unintentionally producing software that is not gender-inclusive. Past research shows that issues of gender-inclusiveness are pervasive in problem-solving software, and until now, software teams like 'John's' at Company X have had no mechanism to find out if their products suffer from such issues, and if so, exactly where the issues are or why they are issues. With GenderMag, we hope that John and others like him will have a tool that helps them head off situations like the one experienced by 'F4', the female end-user programmer in (Cao *et al.*, 2010b):

F4: 'This is so hard for me. Why is it so difficult?

ACKNOWLEDGEMENTS

We thank the participants of our studies for their time; Caitlin Kelleher, Mary Beth Rosson and Rachel White for their help during formative stages of this work; and NSF for their support under CNS-1240957, IIS-1314384, and IIS-1528061.

REFERENCES

Adlin, T. and Pruitt, J. (2010) The Essential Persona Lifecycle: Your Guide to Building and Using Personas. Morgan Kaufmann/Elsevier, Burlington, MA, USA.

Anderson, J.R. (1987) Skill acquisition: compilation of weak-method solutions. Psychol. Rev., 94, 192–211.

Appel, M., Kronberger, N. and Aronson, J. (2011) Stereotype threat impairs ability building: effects on test preparation among women in science and technology. Eur. J. Soc. Psychol., 41, 904–913.

Arjona-Reina, L., Robles, G. and Dueñas, S. (2014) The FLOSS2013 Free/Libre/Open Source Survey.

Bandura, A. (1986) Social Foundations of Thought and Action. Prentice Hall. NJ, USA: Englewood Cliffs.

Bardzell, S. (2010) Feminist HCI: Taking Stock and Outlining an Agenda for Design. In Proc. CHI, pp. 1301–1310. ACM, New York, NY, USA.

Beckwith, L., Burnett, M., Wiedenbeck, S., Cook, C., Sorte, S. and Hasting, M. (2005) Effectiveness of End-user Debugging Software Features: Are there Gender Issues? In Proc. CHI, pp. 869–878. ACM, New York, NY, USA.

Beckwith, L., Kissinger, C., Burnett, M., Wiedenbeck, S., Lawrance, J., Blackwell, A. and Cook, C. (2006) Tinkering and Gender in End-user Programmers' Debugging. In Proc. CHI, pp. 231–240. ACM, New York, NY, USA.

Beckwith, L., Inman, D., Rector, K. and Burnett, M. (2007) On to the Real World: Gender and Self-efficacy in Excel. In Proc. Visual Languages and Human-Centric Computing, pp. 119–126. IEEE, Los Alamitos, CA, USA.

Bell, B., Rieman, J. and Lewis, C. (1991) Usability Testing of a Graphical Programming System: Things We Missed in a Programming Walkthrough. In Proc. CHI, pp. 7–12. ACM, New York, NY, USA.

Berenson, S.B., Slaten, K.M., Williams, L. and Ho, C.-W. (2004) Voices of women in a software engineering course: reflections on collaboration. J. Educ. Resour. Comput., 4, 1.

Blandford, A., Hyde, J., Green, T.R.G. and Connell, I. (2008) Scoping usability evaluation methods: a case study. Hum. Comput. Interact. J., 23, 278–327.

Borkin, M., Yeh, C., Boyd, M., Macko, P., Gajos, K., Seltzer, M. and Pfister, H. (2013) Evaluation of Filesystem Provenance Visualization Tools. In Trans. Vis. Comput. Graphics 19, 12, pp. 2476–2485. IEEE, Los Alamitos, CA, USA.

Bradley, A., MacArthur, C., Hancock, M. and Carpendale, S. (2015) Gendered or Neutral? Considering the Language of HCI. In Proc. Graphics Interface Conference, pp. 163–170. ACM, New York, NY, USA.

Briggs, P., Thomas, L. and Mavin, S. (2014) On Family and Fear: A Gendered Perspective on the Design of Identity Technologies. In CHI 2014 Workshop: Perspectives on Gender and Product Design. https://www.sites.google.com/site/technologydesignperspectives/papers (accessed February 23, 2015).

Burnett, M., Fleming, S., Iqbal, S., Venolia, G., Rajaram, V., Farooq, U., Grigoreanu, V. and Czerwinski, M. (2010) Gender Differences and Programming Environments: Across Programming Populations. In Proc. ACM Empirical Software Engineering and Measurement (ESEM). ACM, New York, NY, USA.

Burnett, M., Beckwith, L., Wiedenbeck, S., Fleming, S., Cao, J., Park, T., Grigoreanu, V. and Rector, K. (2011) Gender pluralism in problem-solving software. Interact. Comput., 23, 450–460.

Burnett, M., Peters, A., Hill, C. and Elarief, N. (2016) Finding Gender-Inclusiveness Software Issues with GenderMag: A Field Investigation. In Proc. CHI. ACM, New York, NY, USA.

Butler, J. (1999) Gender Trouble: Feminism and the Subversion of Identity. Routledge, New York, NY, USA.

Cafferata, P. and Tybout, A. (1989) Gender Differences in Information Processing: A Selectivity Interpretation, Cognitive and Affective Responses to Advertising. Lexington Books, MA, USA.

Cao, J., Rector, K., Park, T., Fleming, S., Burnett, M. and Wiedenbeck, S. (2010a) A Debugging Perspective on End-User Mashup Programming. In Proc. IEEE Visual Languages and Human-Centric Computing, pp. 149–156. IEEE, Los Alamitos, CA, USA.

Cao, J., Riche, Y., Wiedenbeck, S., Burnett, M. and Grigoreanu, V. (2010b) End-User Mashup Programming: Through the Design Lens. In Proc. CHI, pp. 1009–1018. ACM, New York, NY, USA.

Cao, J., Kwan, I., Bahmani, F., Burnett, M., Fleming, S., Jordahl, J., Horvath, A. and Yang, S. (2013) End-User Programmers in Trouble: Can the Idea Garden Help them to Help Themselves? In Proc. Symposium on Visual Languages and Human-Centric Computing. IEEE, Los Alamitos, CA, USA.

Cassell, J. (2002) Genderizing HCI. In Jacko, J. and Sears, A. (eds) The Handbook of Human–Computer Interaction, pp. 402–411. Lawrence Erlbaum, Mahwah, NJ, USA.

Chang, S., Kumar, V., Gilbert, E. and Terveen, L. (2009) Specialization, Homophily, and Gender in a Social Curation Site: Findings from Pinterest. In Proc. ACM Computer Supported Cooperative Work & Social Computing, pp. 674–686. ACM, New York, NY, USA.

Charness, G. and Gneezy, U. (2012) Strong evidence for gender differences in risk taking. J. Econ. Behav. Organ., 83, 50–58.

Churchill, E. (2010) Sugared puppy-dog tails: gender and design. Interactions, 17, 52–56.

Cooper, A. (2004) The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity. Sams Publishing, USA.

Coursaris, C., Swierenga, S. and Watrall, E. (2008) An empirical investigation of color temperature and gender effects on web aesthetics. J. Usability Stud., 3, 103–117.

Davidson, J. and Jensen, C. (2013) Participatory Design with Older Adults: An Analysis of Creativity in the Design of Mobile Healthcare Applications. In Proc. Creativity & Cognition, pp. 114–123. ACM, New York, NY, USA.

Dohmen, T., Falk, A., Huffman, D., Sunde, U., Schupp, J. and Wagner, G.G. (2011) Individual risk attitudes: measurement, determinants, and behavioral consequences. J. Eur. Econ. Assoc., 9, 522–550.

Durndell, A. and Haag, Z. (2002) Computer self-efficacy, computer anxiety, attitudes towards the Internet and reported experience with the Internet, by gender, in an East European sample. Comput. Hum. Behav., 18, 521–535.

Else-Quest, N.M., Hyde, J.S. and Linn, M.C. (2010) Cross-national patterns of gender differences in mathematics: a meta-analysis. Psychol. Bull., 136, 103–127.

ESA (Entertainment Software Association) (2015) Essential Facts About the Computer and Video Game Industry: 2015 Sales, Demographic, and Usage Data. http://www.theesa.com/wp-content/uploads/2015/04/ESA-Essential-Facts-2015.pdf.

Faily, S. and Flechais, I. (2011) Persona Cases: A Technique for Grounding Personas. In Proc. CHI, pp. 2267–2270. ACM, New York, NY, USA.

Friess, E. (2012) Personas and Decision-Making in the Design Process: An Ethnographic Case Study. In Proc. CHI, pp. 1209–1218. ACM, New York, NY, USA.

Greeno, J.G. and Simon, H.A. (1988) Problem Solving and Reasoning. In Atkinson, R.C., Hernstein, R., Lindzey, G. and Luce, R.D. (eds) Stevens Handbook of Experimental Psychology. John Wiley and Sons, NY.

Grigoreanu, V., Cao, J., Kulesza, T., Bogart, C., Rector, K., Burnett, M. and Wiedenbeck, S. (2008) Can Feature Design Reduce the Gender Gap in End-user Software Development Environments? In Proc. Symposium on Visual Languages and Human-Centric Computing, pp. 149–156. IEEE, Los Alamitos, CA, USA.

Grigoreanu, V., Burnett, M. and Robertson, G. (2010) A Strategy-centric Approach to the Design of End-user Debugging Tools. In Proc. CHI, pp. 713–722. ACM, New York, NY, USA.

Grigoreanu, V., Burnett, M., Wiedenbeck, S., Cao, J., Rector, K. and Kwan, I. (2012) End-user debugging strategies: a sensemaking perspective. ACM Trans. Comput.-Hum. Interact., 19, doi:10.1145/2147783.2147788.

Gross, P., Herstand, M., Hodges, J. and Kelleher, C. (2012) A Code Reuse Interface for Non-programmer Middle School Students. In Proceedings Intelligent User Interfaces, pp. 219–228. ACM, New York, NY, USA.

Grudin, J. (2006) Why Personas Work: The Psychological Evidence. In Pruitt, J. and Adlin, T. (eds) The Persona LifeCycle: Keeping People in Mind Throughout Product Design. Morgan Kaufmann Publishers, Burlington, MA, USA.

Hartzel, K. (2003) How Self-efficacy and Gender Issues Affect Software Adoption and Use. In Communications ACM 46, pp. 167–171. ACM, New York, NY, USA.

Hertzum, M., Ebbe, Jacobsen, N. (1999) The Evaluator Effect During First-time Use of the Cognitive Walkthrough Technique. In Proc. HCI International, Vol. I, pp. 1063–1067. Springer, Heidelberg, Germany.

Hou, W., Kaur, M., Komlodi, A., Lutters, W.G., Boot, L., Cotten, S.R., Morrell, C., Ant, Ozok, A. and Tufekci, Z. (2006) 'Girls Don't Waste Time': Pre-adolescent Attitudes Toward ICT. In Proc. CHI Extended Abstracts, pp. 875–880. ACM.

Huffman, A.H., Whetten, J. and Huffman, W.H. (2013) Using technology in higher education: the influence of gender roles on technology self-efficacy. Comput. Hum. Behav., 29, 1779–1786.

Jernigan, W., Horvath, A., Lee, M., Burnett, M., Cuilty, T., Kuttal, S., Peters, A., Kwan, I., Bahmani, F. and Ko, A. (2015) A Principled Evaluation for a Principled Idea Garden. In Proc. IEEE Visual Languages and Human-Centric Computing pp. 235–243. IEEE, Los Alamitos, CA, USA.

Joyce, K., Williamson, J. and Mamo, L. (2007) Technology, science, and ageism: an examination of three patterns of discrimination. Indian J. Gerontol., 21, 110–127.

Kelleher, C., Pausch, R. and Kiesler, S. (2007) Storytelling Alice Motivates Middle School Girls to Learn Computer Programming. In Proc. CHI, pp. 1455–1464. ACM, New York, NY, USA.

Ko, A., Myers, B. and Aung, H. (2004) Six Learning Barriers in End-user Programming Systems. In Proc. Visual Languages and Human-Centric Computing, pp. 199–206. IEEE, Los Alamitos, CA, USA.

Kulesza, T., Stumpf, S., Wong, W.-K., Burnett, M., Perona, S., Ko, A. and Oberst, I. (2011) Why-oriented end-user debugging of naïve bayes text classification. ACM Trans. Interact. Intell. Syst., 1, doi: 10.1145/2030365.2030367.

Lee, M. and Ko, A. (2011) Personifying Programming Tool Feedback Improves Novice Programmers' Learning. In Proc. ICER, pp. 109–116. ACM, New York, NY, USA.

Lee, M., Bahmani, F., Kwan, I., Laferte, J., Charters, P., Horvath, A., Luor, F., Cao, J., Law, C., Bethwetherick, M., Long, S., Burnett, M. and Ko, A. (2014) Principles of

a Debugging-first Puzzle Game for Computing Education. In Proc. Visual Languages and Human-Centric Computing, pp. 57–64. IEEE, Los Alamitos, CA, USA.

Lewis, C., Polson, P., Wharton, C. and Rieman, J. (1990) Testing a Walkthrough Methodology for Theory-based Design of Walk-up-and-Use Interfaces. In Proc. CHI, pp. 235–242. ACM, New York, NY, USA.

Lewis, C., Polson, P. and Rieman, J. (1991) Cognitive walkthrough forms and instructions. Institute of Cognitive Science Technical Report #ICS 91-14. University of Colorado, Boulder, CO, 80309.

Luger, E. (2014) A Design for Life: Recognizing the Gendered Politics Affecting Product Design. In CHI Workshop: Perspectives on Gender and Product Design. https://www.sites.google.com/site/technologydesignperspectives/papers.

Mahatody, T., Sagar, M. and Kolski, C. (2010) State of the art on the Cognitive Walkthrough method, its variants and evolutions. Int. J. HCI, 26, 741–785.

Margolis, J. and Fisher, A. (2003) Unlocking the Clubhouse: Women in Computing. MIT Press, Cambridge, MA, USA.

Marsden, N. (2014) CHI 2014 Workshop on Perspectives on Gender and Product Design. https://www.sites.google.com/site/technologydesignperspectives/papers.

Matthews, T., Judge, T. and Whittaker, S. (2012) How Do Designers and User Experience Professionals Actually Perceive and Use Personas? In Proc. CHI, pp. 1219–1228. ACM, New York, NY, USA.

McDowell, C., Werner, L., Bullock, H.E. and Fernald, J. (2003) The Impact of Pair Programming on Student Performance, Perception and Persistence. In Proc. ICSE. ACM, New York, NY, USA.

McGinn, J. and Kotamraju, N. (2008) Data-driven Persona Development. In Proc. CHI, pp. 1521–1524. ACM, New York, NY, USA.

Meyers-Levy, J. and Loken, B. (2015) Revisiting gender differences: what we know and what lies ahead. J. Consum. Psychol. 25, 129–149.

Meyers-Levy, J. and Maheswaran, D. (1991) Exploring differences in males' and females' processing strategies. J. Consum. Res., 18, 63–70.

Nielsen, L., Storgaard, Hansen, K. (2014) Personas is Applicable: A Study on the use of Personas in Denmark. In Proc. CHI, pp. 1665–1674. ACM, New York, NY, USA.

Norman, D.A. (2002) The Design of Everyday Things. Basic Books, New York.

O'Donnell, E. and Johnson, E.N. (2001) Gender effects on processing effort during analytical procedures. Int. J. Audit., 5, 91–105.

O'Leary-Kelly, A., Hardgrave, B., McKinney, V. and Wilson, D. (2004) The Influence of Professional Identification on the Retention of Women and Racial Minorities in the IT Workforce. NSF ITWF & ITR/EWF Principal Investigator Conference, Philadelphia, PA, pp. 65–69.

Piazza, Blog. (2015) STEM Confidence Gap. http://blog.piazza.com/stem-confidence-gap/.

Polson, P. and Lewis, C. (1990) Theory-based design for easily learned interfaces. Hum. Comput. Interact., 5, 191–220.

Polson, P., Lewis, C., Rieman, J. and Wharton, C. (1992) Cognitive walkthroughs: a method for theory-based evaluation of user interfaces. Int. J. Man-Machine Stud., 36, 741–773.

Power, C., Freire, A., Petrie, H. and Swallow, D. (2012) Guidelines are Only Half of the Story: Accessibility Problems Encountered by Blind users on the Web. In Proc. CHI, pp. 433–442. ACM, New York, NY, USA.

Pruitt, J. and Grudin, J. (2003) Personas: Practice and Theory. In Proc. DUX, pp. 1–15. ACM, New York, NY, USA.

Riedl, R., Hubert, M. and Kenning, P. (2010) Are there neural gender differences in online trust? An fMRI study on the perceived trustworthiness of ebay offers. MIS Q., 34, 397–428.

Rode, J. (2008) An ethnographic examination of the relationship of gender & end-user programming. PhD Thesis, University of California Irvine.

Rosner, D. and Bean, J. (2009) Learning from IKEA Hacking: I'm Not One to Decoupage a Tabletop and Call it a Day. In Proc. CHI, pp. 419–422. ACM, New York, NY, USA.

Rosson, M., Sinha, H., Bhattacharya, M. and Zhao, D. (2007) Design Planning in End-user Web Development. In Proc. Visual Languages and Human-Centric Computing, pp. 189–196. IEEE, Los Alamitos, CA, USA.

Rosson, M., Sinha, H. and Edor, T. (2010) Design Planning in End-user Web Development: Gender, Feature Exploration, and Feelings of Success. In Proc. Visual Languages and Human-Centric Computing, pp. 141–148. IEEE, Los Alamitos, CA, USA.

Simon, S.J. (2001) The impact of culture and gender on web sites: an empirical study. Data Base Adv. Inf. Syst., 32, 18–37.

Singh, A., Bhadauria, V., Jain, A. and Gurung, A. (2013) Role of gender, self-efficacy, anxiety and testing formats in learning spreadsheets. Comput. Hum. Behav., 29, 739–746.

Spencer, R. (2000) The Streamlined Cognitive Walkthrough Method, Working Around Social Constraints Encountered in a Software Development Company. In Proc. CHI, pp. 353–359. ACM, New York, NY, USA.

Stumpf, S., Sullivan, E., Fitzhenry, E., Oberst, I., Wong, W.-K. and Burnett, M. (2008) Integrating Rich User Feedback into Intelligent User Interfaces. In Proc. Intelligent User Interfaces, pp. 50–59. ACM, New York, NY, USA.

Subrahmaniyan, N., Beckwith, L., Grigoreanu, V., Burnett, M., Wiedenbeck, S., Narayanan, V., Bucht, K., Drummond, R. and Fern, X. (2008) Testing vs. Code Inspection vs. … What Else? Male and Female end Users' Debugging Strategies. In Proc. CHI, pp. 617–626. ACM, New York, NY, USA.

Szafir, D. and Mutlu, B. (2012) Pay Attention! Designing Adaptive Agents that Monitor and Improve user Engagement. In Proc. CHI, pp. 11–20. ACM, New York, NY, USA.

Tan, D., Czerwinski, M. and Robertson, G. (2003) Women Go with the (Optical) Flow. In Proc. CHI, pp. 209–215. ACM, New York, NY, USA.

Turner, P. and Turner, S. (2011) Is stereotyping inevitable when designing with personas? Des. Stud., 32, 30–44.

Weber, E., Blais, A. and Betz, N. (2002) A domain-specific risk-attitude scale: measuring risk perceptions and risk behaviors. J. Behav. Decis. Mak., 15, 263–290.

West, C. and Zimmerman, D.H. (1987) Doing gender. Gender Soc., 1, 125–151.

Wharton, C., Bradford, J., Jeffries, R. and Franzke, M. (1992) Applying Cognitive Walkthroughs to More Complex user Interfaces: Experiences, Issues, and Recommendations. In Proc. CHI, pp. 381–388. ACM, New York, NY, USA.

Wharton, C., Rieman, J., Lewis, C. and Polson, P. (1994) The Cognitive Walkthrough Method: A Practitioner's Guide. In Nielsen. J. and Mack R. (eds) Usability Inspection Methods, pp. 105–140. John Wiley, NY.

Williams, G. (2014) Are you sure your software is gender-neutral? Interactions, 21, 36–39.

Zhang, S., Schmader, T. and Hall, W.M. (2013) L'eggo my ego: Reducing the gender gap in math by unlinking the self from performance. Self Identity, 12, 400–412.

APPENDIX
THE FOUR PERSONAS

Tim (Timothy) Hopkins

- 28 years old
- Employed as an Accountant
- Lives in Cardiff, Wales

Tim loves public transportation. He knows several routes to get there from home and he's always exploring ways to optimize his trips into the office. Work starts with email, which he answers one at a time, as soon as he reads them. (Sometimes this backfires, if there is a second related message he hasn't read yet.) Some nights he plays computer games with his online friends.

Background knowledge and skills

- Tim works as an accountant in a consulting firm. He just moved to this employer 1 week ago, and their software systems are new to him. He describes himself as a "numbers person". He is not a professional programmer but he writes and edits spreadsheet formulas in his work.
- Tim has a degree in accounting so he knows plenty of Maths and knows how to think in terms of numbers. He's never taken any computer programming classes.
- He plays the latest video games, has the newest smart phone and a hybrid car. He downloads and installs the latest software, and experiments with its settings. He is comfortable and confident with technology and he enjoys learning about it and using new technologies.

Motivations and Strategies

- ***Motivations*:** Tim is proficient with the technologies he uses. He likes learning all the available functionality on all of his devices and all computer systems he uses, even when it may not be necessary help him achieve his tasks. He likes trying to figure out if there is more than one way of doing the same thing. He sometimes finds himself exploring functions of one of his gadgets for so long that he loses sight of what he wanted to do with it to begin with.
- ***Information Processing Style:*** When he is exploring or searching for information to solve a problem with a computer program or gadget he uses, he likes a *selective information processing style* or "depth first" approach. That is, he usually delves into the first promising option, pursues it, and if it doesn't work out he backs out and gathers a bit more information until he sees another option to try. Thus, his style is very incremental.

Attitude to Technology

For Tim, technology is a source of fun, and he is always on the lookout for new computer software. He likes to make sure he has the latest version of all software with all the new features.

- ***Computer Self-Efficacy*:** Tim is pretty confident in his abilities with technology, and thinks he's better than the average person at learning about new features. He doesn't give up easily when computers or technology present a challenge to him. He will spend hours reading the manual so he can configure something to do exactly what he wants to do. If he can't fix the problem, he blames it on the software vendor; it's not his fault if he can't get it to work.
- ***Attitude toward Risk:*** Tim doesn't mind taking risks using features of technology that haven't been proven to work. When he is presented with challenges because he has tried a new way that doesn't work, it doesn't change his attitudes toward technology.
- ***Willingness to Explore and Tinker*:** Whenever Tim uses new technology, he tries to construct his own understanding of how the software works internally. He likes tinkering and exploring the menu items and functions of the software or device in order to build that understanding. Sometimes though he plays with features that make things not work so well.

Figure A1. The Tim persona, representing users with facet values most common in males, as in Fig. 1's 'Value A'. The five facets are bulleted in the bottom two rounded rectangles. The red, underlined parts are to enable the evaluation team to quickly remind themselves of the main points.

Abby (Abigail) Jones

- **28 years old**
- **Employed as an Accountant**
- **Lives in Cardiff, Wales**

When Abby drives to work in the mornings, she listens to her favorite music. She likes a variety of music, and adds to her music collection often. When she arrives at work, she scans all her emails first to get an overall picture before answering any of them. (This extra pass takes time but seems worth it.) Some nights she goes to yoga classes and plays computer puzzle games.

Background knowledge and skills

- Abby works as an accountant in a consulting firm. She just moved to this employer 1 week ago, and their software systems are new to her. She describes herself as a "numbers person". She is not a professional programmer but she writes and edits spreadsheet formulas in her work.
- Abby has a degree in accounting, so she knows plenty of Maths and knows how to think in terms of numbers. She's never taken any computer programming or IT systems classes.
- Even though she's an accountant and deals with numbers all day at work, she enjoys working with numbers and logic in her free time, too. She especially likes working puzzles, either on paper or on the computer, such as Sudoku and other puzzle games.

Motivations and Strategies

- ***Motivations*:** Abby is proficient with the technologies she uses. She learns new technologies when she needs to, but she doesn't spend her free time exploring technology or exploring obscure functionality of programs and devices that she uses. When Abby uses computers to problem-solve, she has little desire to learn new functions, or to search for information on them. She tries to use methods she is already familiar and comfortable with to achieve her goals.
- ***Information Processing Style:*** Abby leans towards a *comprehensive information processing style* when she needs to gather information to problem-solve. That is, before following any option that seems promising, she first gathers information comprehensively to try to form a complete understanding of the problem before trying to solve it. Thus, her style is "burst-y"; first she reads a lot, then she acts on it in a batch of activity.

Attitude to Technology

Abby is generally comfortable using familiar technology, but she does not get a big kick out of obtaining the latest gadgets or learning how to use them. She prefers to stay with the technologies for which she has already mastered the peculiarities.

- ***Computer Self-Efficacy*:** Abby has low computer self-efficacy, meaning that she has low self-confidence in performing computing tasks other than the ones she is familiar with. This has a variety of impacts on how she uses software. For example, she is not confident that she can learn to use new features and, as self-efficacy theory explains, she often gives up if she runs into challenges. Software with usability problems poses more challenges to her than it does to more confident users, and she often blames herself for problems that she encounters.
- ***Attitude toward Risk:*** Abby is risk averse when she uses computers to perform tasks. When confronted with new software features, Abby worries that she will spend time on them and not get any benefits from doing so. She tries to perform tasks "the safe" (ie, familiar) way, even if the less familiar features might promise a more direct solution.
- ***Willingness to Explore and Tinker*:** Abby doesn't particularly like tinkering (ie, just trying out new features or commands to see what they do) when she uses software in her work tasks. She prefers following step-by-step tutorials and wizards over tinkering. (However, when she does tinker, it has positive effects on her understanding of the software.) When software features/commands cause problems for her, she tends to blame herself and, if she can, she then just avoids those troublesome features/commands. She then uses work-arounds that involve using only features/commands she is familiar with already.

Figure A2. The Abby persona, representing female users with facet values most dissimilar to Tim's, as in 'Value C'.

Pat (Patricia) Jones

- 43 years old
- Employed as an Accountant
- Lives in Cardiff, Wales

Pat loves public transportation and knows at least three routes to get there from home. When she arrives at work, she scans all her emails first to get an overall picture before answering any of them. (This extra pass takes time but seems worth it.) Some evenings she plays computer puzzle games like Sudoku before bed.

Background knowledge and skills

- Pat works as an accountant in a consulting firm. She just moved to this employer 1 week ago, and their software systems are new to her. She describes herself as a "numbers person". She is not a professional programmer but she writes and edits spreadsheet formulas in her work.
- Pat has a degree in accounting, so she knows plenty of Maths and knows how to think in terms of numbers. She's never taken any computer programming or IT systems classes.
- Even though she's an accountant and deals with numbers all day at work, she likes working with numbers in her free time, too. She especially likes Sudoku and other computer games that involve puzzling.

Motivations and Strategies

- ***Motivations*:** Pat is proficient with the technologies she uses. She learns new technologies when she needs to, but she doesn't spend her free time exploring technology or exploring obscure functionality of programs and devices that she uses. She tends to use methods she is already familiar and comfortable with to achieve her goals.
- ***Information Processing Style:*** Pat leans towards a *comprehensive information processing style* when she needs to gather information to problem-solve. That is, before following any option that seems promising, she first gathers information comprehensively to try to form a complete understanding of the problem before trying to solve it. Thus, her style is "burst-y"; first she reads a lot, then she acts on it in a batch of activity.

Attitude to Technology

Pat is generally comfortable using familiar technology, but she does not get a big kick out of obtaining the latest gadgets or learning how to use them. She prefers to stay with the technologies for which she has already mastered the peculiarities.

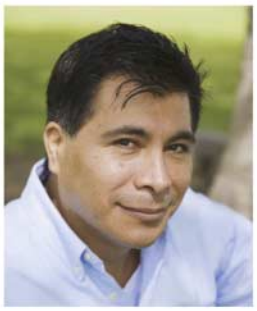
- ***Computer Self-Efficacy*:** Pat has medium computer self-efficacy, meaning that she has some self-confidence in performing computing tasks other than the ones she is familiar with. This has a variety of impacts on how she uses software. For example, she will keep on trying to figure out how to achieve what she has set out to do for quite awhile; she doesn't give up right away when computers or technology present a challenge to her.
- ***Attitude toward Risk:*** Even so, Pat is risk averse when she uses computers to perform tasks. When confronted with new software features, Pat worries that she will spend time on them and not get any benefits from doing so. She prefers to perform tasks "the safe" (ie, familiar) way if possible, even if less familiar features might promise a more direct solution.
- ***Willingness to Explore and Tinker*:** When Pat sees a need to learn new technology, she does so by trying out new features or commands to see what they do and to understand how the software works. When she does this, she does so purposefully; that is, she reflects on each bit of feedback she gets along the way to understand how the feature might benefit her. Eventually, if she doesn't think it will get her closer to what she wants to achieve, she will revert back to ways that she already knows will work.

Figure A3. The Patricia persona, representing female users with most values along the lines of 'Value B'. Patricia is identical to Patrick (Fig. A4) except for her gender.

Pat (Patrick) Jones

- **43 years old**
- **Employed as an Accountant**
- **Lives in Cardiff, Wales**

Pat loves public transportation and knows at least three routes to get there from home. When he arrives at work, he scans all his emails first to get an overall picture before answering any of them. (This extra pass takes time but seems worth it.) Some evenings he plays computer puzzle games like Sudoku before bed.



Background knowledge and skills

- Pat works as an accountant in a consulting firm. He just moved to this employer 1 week ago, and their software systems are new to him. He describes himself as a "numbers person". He is not a professional programmer but he writes and edits spreadsheet formulas in her work.
- Pat has a degree in accounting so he knows plenty of Maths and knows how to think in terms of numbers. He's never taken any computer programming or IT systems classes.
- Even though he's an accountant and deals with numbers all day at work, he likes working with numbers in his free time, too. He especially likes Sudoku and other computer games that involve puzzling.

Motivations and Strategies

- ***Motivations*:** Pat is proficient with the technologies he uses. He learns new technologies when he needs to, but he doesn't spend his free time exploring technology or exploring obscure functionality of programs and devices that he uses. He tends to use methods he is already familiar and comfortable with to achieve his goals.
- ***Information Processing Style:*** Pat leans towards a *comprehensive information processing style* when he needs to gather information to problem-solve. That is, before following some option that seems promising, he first gathers information comprehensively to try to form a complete understanding of the problem before trying to solve it. Thus, his style is "burst-y"; first he reads a lot, then he acts on it in a batch of activity.

Attitude to Technology

Pat is generally comfortable using familiar technology, but he does not get a big kick out of obtaining the latest gadgets or learning how to use them. He prefers to stay with the technologies for which he has already mastered the peculiarities.

- ***Computer Self-Efficacy*:** Pat has medium computer self-efficacy, meaning that he has some self-confidence in performing computing tasks other than the ones he is familiar with. This has a variety of impacts on how he uses software. For example, he will keep on trying to figure out how to achieve what he has set out to do for quite awhile; he doesn't give up right away when computers or technology present a challenge to him.
- ***Attitude toward Risk:*** Even so, Pat is risk averse when he uses computers to perform tasks. When confronted with new software features, Pat worries that he will spend time on them and not get any benefits from doing so. He prefers to perform tasks "the safe" (ie, familiar) way if possible, even if less familiar features might promise a more direct solution.
- ***Willingness to Explore and Tinker*:** When Pat sees a need to learn new technology, he does so by trying out new features or commands to see what they do and to understand how the software works. When he does this, he does so purposefully; that is, he reflects on each bit of feedback he gets along the way to understand how the feature might benefit him. Eventually, if he doesn't think it will get him closer to what he wants to achieve, he will revert back to ways that he already knows will work.

Figure A4. The Patrick persona, representing male users with most values along the lines of 'Value B'. Patrick is identical to Patricia except for his gender.