

Margaret Burnett

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/12045228/publications.pdf>

Version: 2024-02-01

80
papers

3,162
citations

516710

16
h-index

454955

30
g-index

82
all docs

82
docs citations

82
times ranked

1209
citing authors

#	ARTICLE	IF	CITATIONS
1	How Gender-Biased Tools Shape Newcomer Experiences in OSS Projects. IEEE Transactions on Software Engineering, 2022, 48, 241-259.	5.6	18
2	How Do People Rank Multiple Mutant Agents?. , 2022, , .		2
3	Finding AI's Faults with AAR/AI: An Empirical Study. ACM Transactions on Interactive Intelligent Systems, 2022, 12, 1-33.	3.7	4
4	How to Debug Inclusivity Bugs? A Debugging Process with Information Architecture. , 2022, , .		4
5	The Shoutcasters, the Game Enthusiasts, and the AI: Foraging for Explanations of Real-time Strategy Players. ACM Transactions on Interactive Intelligent Systems, 2021, 11, 1-46.	3.7	2
6	AID: An Automated Detector for Gender-Inclusivity Bugs in OSS Project Pages. , 2021, , .		10
7	From "no clear winner" to an effective Explainable Artificial Intelligence process: An empirical journey. Applied AI Letters, 2021, 2, .	2.2	5
8	Gender Inclusivity as a Quality Requirement: Practices and Pitfalls. IEEE Software, 2020, 37, 7-11.	1.8	11
9	Keeping it "organized and logical". , 2020, , .		9
10	Engineering gender-inclusivity into software. , 2020, , .		17
11	Doing Inclusive Design. , 2020, , .		3
12	Special Issue on Highlights of ACM Intelligent User Interface (IUI) 2018. ACM Transactions on Interactive Intelligent Systems, 2020, 10, 1-3.	3.7	0
13	From Gender Biases to Gender-Inclusive Design. , 2019, , .		72
14	How end-user programmers debug visual web-based programs: An information foraging theory perspective. Journal of Computer Languages, 2019, 53, 22-37.	2.1	9
15	From GenderMag to InclusiveMag: An Inclusive Design Meta-Method. , 2019, , .		15
16	Pedagogical Content Knowledge for Teaching Inclusive Design. , 2018, , .		20
17	The GenderMag Recorder's Assistant. , 2018, , .		1
18	Semi-Automating (or not) a Socio-Technical Method for Socio-Technical Systems. , 2018, , .		6

#	ARTICLE	IF	CITATIONS
19	Gender in open source software. , 2018, , .		15
20	Open source barriers to entry, revisited. , 2018, , .		82
21	How the Experts Do It. , 2018, , .		20
22	Gender-Inclusiveness Personas vs. Stereotyping. , 2017, , .		57
23	Toward Theory-Based End-User Software Engineering. , 2017, , 231-268.		1
24	Trials and tribulations of developers of intelligent systems: A field study. , 2016, , .		66
25	Foraging and navigations, fundamentally: developers' predictions of value and cost. , 2016, , .		20
26	"Womonomics" and gender-inclusive software: what software engineers need to know (invited talk). , 2016, , .		1
27	GenderMag: A Method for Evaluating Software's Gender Inclusiveness. Interacting With Computers, 2016, 28, 760-787.	1.5	137
28	Finding Gender-Inclusiveness Software Issues with GenderMag. , 2016, , .		65
29	A principled evaluation for a principled idea garden. , 2015, , .		22
30	To fix or to learn? How production bias affects developers' information foraging during debugging. , 2015, , .		26
31	Principles of Explanatory Debugging to Personalize Interactive Machine Learning. , 2015, , .		258
32	Idea Garden: Situated Support for Problem Solving by End-User Programmers. Interacting With Computers, 2015, 27, 640-660.	1.5	12
33	Principles of a debugging-first puzzle game for computing education. , 2014, , .		87
34	You Are the Only Possible Oracle: Effective Test Selection for End Users of Interactive Machine Learning Systems. IEEE Transactions on Software Engineering, 2014, 40, 307-323.	5.6	44
35	How Programmers Debug, Revisited: An Information Foraging Theory Perspective. IEEE Transactions on Software Engineering, 2013, 39, 197-215.	5.6	97
36	End-user programmers in trouble: Can the Idea Garden help them to help themselves?. , 2013, , .		8

#	ARTICLE	IF	CITATIONS
37	Reactive information foraging. , 2012, , .		37
38	End-user debugging strategies. ACM Transactions on Computer-Human Interaction, 2012, 19, 1-28.	5.7	45
39	From barriers to learning in the idea garden: An empirical study. , 2012, , .		10
40	End-User Software Engineering and Why It Matters. , 2012, , 185-201.		1
41	An exploration of design opportunities for “gardening” end-user programmers' ideas. , 2011, , .		12
42	The state of the art in end-user software engineering. ACM Computing Surveys, 2011, 43, 1-44.	23.0	407
43	Mini-crowdsourcing end-user assessment of intelligent assistants: A cost-benefit study. , 2011, , .		11
44	Gender differences and programming environments. , 2010, , .		65
45	End-user mashup programming. , 2010, , .		34
46	A Debugging Perspective on End-User Mashup Programming. , 2010, , .		48
47	End-User Software Engineering and Why it Matters. Journal of Organizational and End User Computing, 2010, 22, 1-22.	2.9	6
48	End-user strategy programming. Journal of Visual Languages and Computing, 2009, 20, 16-29.	1.8	12
49	What Is End-User Software Engineering and Why Does It Matter?. Lecture Notes in Computer Science, 2009, , 15-28.	1.3	36
50	Can feature design reduce the gender gap in end-user software development environments?. Visual Languages and Human-Centric Computing, 2009 VL/HCC 2009 IEEE Symposium on, 2008, , .	0.0	37
51	Testing vs. code inspection vs. what else?. , 2008, , .		38
52	Integrating rich user feedback into intelligent user interfaces. , 2008, , .		27
53	Using information scent to model the dynamic foraging behavior of programmers in maintenance tasks. , 2008, , .		59
54	Scents in Programs:Does Information Foraging Theory Apply to Program Maintenance?. , 2007, , .		25

#	ARTICLE	IF	CITATIONS
55	Mining Interpretable Human Strategies: A Case Study. , 2007, , .		5
56	On to the Real World: Gender and Self-Efficacy in Excel. , 2007, , .		18
57	Gender HCI: What About the Software?. Computer, 2006, 39, 97-101.	1.1	57
58	Impact of high-intensity negotiated-style interruptions on end-user debugging. Journal of Visual Languages and Computing, 2006, 17, 187-202.	1.8	3
59	Using cognitive dimensions: Advice from the trenches. Journal of Visual Languages and Computing, 2006, 17, 302-327.	1.8	22
60	Tinkering and gender in end-user programmers' debugging. , 2006, , .		108
61	Interactive, visual fault localization support for end-user programmers. Journal of Visual Languages and Computing, 2005, 16, 3-40.	1.8	31
62	Effectiveness of end-user debugging software features. , 2005, , .		75
63	An empirical study of fault localization for end-user programmers. , 2005, , .		24
64	Six challenges in supporting end-user debugging. , 2005, , .		5
65	Six challenges in supporting end-user debugging. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2005, 30, 1-6.	0.7	8
66	End-user software engineering. Communications of the ACM, 2004, 47, 53-58.	4.5	151
67	End-User Testing for the Lye Methodology using the Screen Transition Paradigm and WYSIWYT. Knowledge-Based Systems, 2003, 16, 431-440.	7.1	1
68	HCI research regarding end-user requirement specification: a tutorial. Knowledge-Based Systems, 2003, 16, 341-349.	7.1	3
69	A user-centred approach to functions in excel. , 2003, , .		78
70	Harnessing curiosity to increase correctness in end-user programming. , 2003, , .		48
71	A user-centred approach to functions in Excel. ACM SIGPLAN Notices, 2003, 38, 165-176.	0.2	34
72	A scalable method for deductive generalization in the spreadsheet paradigm. ACM Transactions on Computer-Human Interaction, 2002, 9, 253-284.	5.7	4

#	ARTICLE	IF	CITATIONS
73	End-User Programming of Time as an "Ordinary"™ Dimension in Grid-Oriented Visual Programming Languages. Journal of Visual Languages and Computing, 2002, 13, 421-447.	1.8	4
74	Adding Apples and Oranges. Lecture Notes in Computer Science, 2002, , 173-191.	1.3	44
75	Forms/3: A first-order visual language to explore the boundaries of the spreadsheet paradigm. Journal of Functional Programming, 2001, 11, 155-206.	0.8	127
76	A methodology for testing spreadsheets. ACM Transactions on Software Engineering and Methodology, 2001, 10, 110-147.	6.0	113
77	Slicing spreadsheets. ACM SIGPLAN Notices, 2000, 35, 25-38.	0.2	6
78	Slicing spreadsheets. , 1999, , .		23
79	A bug's eye view of immediate visual feedback in direct-manipulation programming systems. , 1997, , .		7
80	Visually testing recursive programs in spreadsheet languages. , 0, , .		7