Tracy Hall

List of Publications by Year in descending order

90 papers	4,251 citations	218677 26 h-index	214800 47 g-index
92 all docs	92 docs citations	92 times ranked	2259 citing authors

#	Article	IF	Citations
1	A Systematic Literature Review on Fault Prediction Performance in Software Engineering. IEEE Transactions on Software Engineering, 2012, 38, 1276-1304.	5.6	801
2	Motivation in Software Engineering: A systematic literature review. Information and Software Technology, 2008, 50, 860-878.	4.4	337
3	Researcher Bias: The Use of Machine Learning in Software Defect Prediction. IEEE Transactions on Software Engineering, 2014, 40, 603-616.	5.6	252
4	Some Code Smells Have a Significant but Small Effect on Faults. ACM Transactions on Software Engineering and Methodology, 2014, 23, 1-39.	6.0	208
5	De-motivators for software process improvement: an analysis of practitionersâ€™ views. Journal of Systems and Software, 2003, 66, 23-33.	4.5	171
6	Key success factors for implementing software process improvement: a maturity-based analysis. Journal of Systems and Software, 2002, 62, 71-84.	4.5	147
7	Code Bad Smells: a review of current knowledge. Journal of Software: Evolution and Process, 2011, 23, 179-202.	1.1	137
8	Models of motivation in software engineering. Information and Software Technology, 2009, 51, 219-233.	4.4	122
9	Software defect prediction: do different classifiers find the same defects?. Software Quality Journal, 2018, 26, 525-552.	2.2	122
10	Motivators of Software Process Improvement: an analysis of practitioners' views. Journal of Systems and Software, 2002, 62, 85-96.	4.5	116
11	Implementing effective software metrics programs. IEEE Software, 1997, 14, 55-65.	1.8	115
12	Using an expert panel to validate a requirements process improvement model. Journal of Systems and Software, 2005, 76, 251-275.	4.5	114
13	Trust in software outsourcing relationships: An empirical investigation of Indian software companies. Information and Software Technology, 2006, 48, 345-354.	4.4	108
14	Software Process Improvement Problems in Twelve Software Companies: An Empirical Analysis. Empirical Software Engineering, 2003, 8, 7-42.	3.9	107
15	Implementing software process improvement: an empirical study. Software Process Improvement and Practice, 2002, 7, 3-15.	1.1	79
16	A quantitative and qualitative analysis of factors affecting software processes. Journal of Systems and Software, 2003, 66, 7-21.	4.5	74
17	Defining a Requirements Process Improvement Model. Software Quality Journal, 2005, 13, 247-279.	2.2	61
18	Automatically identifying code features for software defect prediction: Using AST N-grams. Information and Software Technology, 2019, 106, 142-160.	4.4	60

#	Article	IF	Citations
19	Factors that motivate software engineering teams: A four country empirical study. Journal of Systems and Software, 2014, 92, 115-127.	4.5	51
20	What Do We Know about Developer Motivation?. IEEE Software, 2008, 25, 92-94.	1.8	46
21	A systematic review of theory use in studies investigating the motivations of software engineers. ACM Transactions on Software Engineering and Methodology, 2009, 18, 1-29.	6.0	46
22	Difficulties in Managing Offshore Software Outsourcing Relationships: An Empirical Analysis of 18 High Maturity Indian Software Companies. Journal of Information Technology Case and Application Research, 2005, 7, 25-41.	0.8	45
23	A framework for evaluation and prediction of software process improvement success. Journal of Systems and Software, 2001, 59, 135-142.	4.5	41
24	The impact of staff turnover on software projects. , 2008, , .		41
25	Comparing the performance of fault prediction models which report multiple performance measures. , 2012, , .		36
26	Mutation-aware fault prediction. , 2016, , .		34
27	Ethical Issues in Software Engineering Research: A Survey of Current Practice. Empirical Software Engineering, 2001, 6, 305-317.	3.9	33
28	Perceptions of software quality: a pilot study. Software Quality Journal, 1998, 7, 67-75.	2.2	31
29	The jinx on the NASA software defect data sets. , 2016, , .		31
30	Building an Ensemble for Software Defect Prediction Based on Diversity Selection. , 2016, , .		30
31	Does the XP environment meet the motivational needs of the software developer? An empirical study. , 2007, , .		27
32	On The Introduction of Automatic Program Repair in Bloomberg. IEEE Software, 2021, 38, 43-51.	1.8	26
33	SLuRp. , 2012, , .		25
34	Software developer motivation in a high maturity company: a case study. Software Process Improvement and Practice, 2006, 11, 219-228.	1.1	23
35	What is the Impact of Imbalance on Software Defect Prediction Performance?. , 2015, , .		23
36	The relationship between evolutionary coupling and defects in large industrial software. Journal of Software: Evolution and Process, 2017, 29, e1842.	1.6	23

#	Article	IF	Citations
37	How Good Are My Tests?. , 2017, , .		23
38	Practitioner roles in software process improvement: an analysis using grid technique. Software Process Improvement and Practice, 2002, 7, 17-31.	1.1	22
39	Developing Fault-Prediction Models: What the Research Can Show Industry. IEEE Software, 2011, 28, 96-99.	1.8	22
40	Mining Communication Patterns in Software Development. , 2018, , .		22
41	DConfusion: a technique to allow cross study performance evaluation of fault prediction studies. Automated Software Engineering, 2014, 21, 287-313.	2.9	21
42	Design Patterns and Change Proneness: A Replication Using Proprietary C# Software. , 2009, , .		20
43	Reproducibility and replicability of software defect prediction studies. Information and Software Technology, 2018, 99, 148-163.	4.4	20
44	Can Thomas Kuhn's paradigms help us understand software engineering?. European Journal of Information Systems, 2004, 13, 235-243.	9.2	19
45	Software process simulation modelling: A survey of practice. Journal of Simulation, 2008, 2, 91-102.	1.5	19
46	Software Process Improvement Motivators: An Analysis using Multidimensional Scaling. Empirical Software Engineering, 2002, 7, 93-114.	3.9	18
47	A Critical Analysis of Current OO Design Metrics. Software Quality Journal, 1999, 8, 97-110.	2.2	17
48	An analysis of some ?core studies? of software process improvement. Software Process Improvement and Practice, 2001, 6, 169-187.	1.1	17
49	The State of Machine Learning Methodology in Software Fault Prediction. , 2012, , .		16
50	Critical factors in software outsourcing. , 2004, , .		15
51	Motivating developer performance to improve project outcomes in a high maturity organization. Software Quality Journal, 2007, 15, 365-381.	2.2	15
52	Prioritising Refactoring Using Code Bad Smells. , 2011, , .		14
53	Simulating global software evolution processes by combining simple models: an initial study. Software Process Improvement and Practice, 2002, 7, 113-126.	1.1	13
54	So You Need More Method Level Datasets for Your Software Defect Prediction?. , 2016, , .		12

#	Article	IF	Citations
55	How Software Developers Mitigate Their Errors When Developing Code. IEEE Transactions on Software Engineering, 2022, 48, 1853-1867.	5.6	12
56	Implementing software metrics ? the critical success factors. Software Quality Journal, 1994, 3, 195-208.	2.2	10
57	Improving the Precision of Fowler's Definitions of Bad Smells. , 2008, , .		9
58	Filling the Gaps of Development Logs and Bug Issue Data. , 2014, , .		9
59	Different Classifiers Find Different Defects Although With Different Level of Consistency. , 2015, , .		9
60	Title is missing!. Software Quality Journal, 1998, 7, 67-75.	2.2	9
61	An initial investigation of software practitioners' motivation. , 2009, , .		8
62	Fault Analysis in OSS Based on Program Slicing Metrics. , 2009, , .		8
63	Evolutionary coupling measurement: Making sense of the current chaos. Science of Computer Programming, 2017, 135, 4-19.	1.9	8
64	How Effectively Is Defective Code Actually Tested?. , 2018, , .		8
65	Authorsâ€™ Reply to â€œComments on â€˜Researcher Bias: The Use of Machine Learning in Software Defect Predictionâ€™â€•. IEEE Transactions on Software Engineering, 2018, 44, 1129-1131.	5.6	7
66	Evaluating Three Approaches to Extracting Fault Data from Software Change Repositories. Lecture Notes in Computer Science, 2010, , 107-115.	1.3	7
67	Software engineering practice versus evidence-based software engineering research. , 2005, , .		6
68	Exploring motivational differences between software developers and project managers. , 2007, , .		6
69	Getting the Best out of Software Process Simulation and Empirical Research in Software Engineering. , 2007, , .		6
70	Do bad smells indicate "trouble" in code?. , 2008, , .		6
71	Software engineering group work. , 2010, , .		6
72	Introduction to Stand-up Meetings in Agile Methods. , 2009, , .		5

#	Article	IF	Citations
73	The impact of media selection on stakeholder communication in agile global software development. , 2011, , .		5
74	The inconsistent measurement of Message Chains. , 2013, , .		5
75	Getting Defect Prediction Into Industrial Practice: the ELFF Tool. , 2017, , .		5
76	Using experimental games to understand communication and trust in Agile software teams. , 2013, , .		4
77	Expanding Fix Patterns to Enable Automatic Program Repair. , 2021, , .		4
78	Software quality programmes: a snapshot of theory versus reality. Software Quality Journal, 1996, 5, 235-242.	2.2	3
79	Reducing Regression Test Size by Exclusion. Conference on Software Maintenance, Proceedings of the, 2007, , .	0.0	2
80	Assert Use and Defectiveness in Industrial Code. , 2017, , .		2
81	Exploiting fault localisation for efficient program repair. , 2020, , .		1
82	Which Software Faults Are Tests Not Detecting?. , 2020, , .		1
83	Using multi dimensional scaling to analyse software engineers' deâ€motivators for SPI. Software Process Improvement and Practice, 2007, 12, 511-522.	1.1	0
84	Program slicing-based cohesion measurement. , 2011, , .		0
85	Introduction to the Special Section from the Empirical Track of the XP2016 conference. Information and Software Technology, 2017, 92, 222.	4.4	0
86	The relationship between evolutionary coupling and defects in large industrial software (journal-first abstract). , 2018, , .		0
87	Code Cleaning for Software Defect Prediction: A Cautionary Tale. , 2018, , .		0
88	Zones of Pain: Visualising theÂRelationship Between Software Architecture and Defects. Communications in Computer and Information Science, 2020, , 135-143.	0.5	0
89	Fault-insertion and fault-fixing: analysing developer activity over time. , 2020, , .		0
90	Using Machine Learning toÂRecognise Novice andÂExpert Programmers. Lecture Notes in Computer Science, 2021, , 199-206.	1.3	0