JÃ¼rgen BÃ¶rstler

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/7486914/publications.pdf

Version: 2024-02-01

69 papers	1,178 citations	687220 13 h-index	580701 25 g-index
70 all docs	70 docs citations	70 times ranked	839 citing authors

#	Article	IF	Citations
1	Educational Data Mining and Learning Analytics in Programming. , 2015, , .		239
2	Taxonomies in software engineering: A Systematic mapping study and a revised taxonomy development method. Information and Software Technology, 2017, 85, 43-59.	3.0	107
3	The PLUSS Approach â€“ Domain Modeling with Features, Use Cases and Use Case Realizations. Lecture Notes in Computer Science, 2005, , 33-44.	1.0	80
4	Empirical evidence on the link between object-oriented measures and external quality attributes: a systematic literature review. Empirical Software Engineering, 2015, 20, 640-693.	3.0	74
5	Effort estimation in agile software development. , 2015, , .		48
6	The impacts of agile and lean practices on project constraints: A tertiary study. Journal of Systems and Software, 2016, 119, 162-183.	3.3	41
7	Teaching PSP: challenges and lessons learned. IEEE Software, 2002, 19, 42-48.	2.1	37
8	Effort estimation in large-scale software development: An industrial case study. Information and Software Technology, 2018, 99, 21-40.	3.0	32
9	Managing requirements specifications for product lines â€“ An approach and industry case study. Journal of Systems and Software, 2009, 82, 435-447.	3.3	31
10	The Role of Method Chains and Comments in Software Readability and Comprehensionâ€”An Experiment. IEEE Transactions on Software Engineering, 2016, 42, 886-898.	4.3	26
11	A controlled empirical evaluation of a requirements abstraction model. Information and Software Technology, 2007, 49, 790-805.	3.0	24
12	An Effort Estimation Taxonomy for Agile Software Development. International Journal of Software Engineering and Knowledge Engineering, 2017, 27, 641-674.	0.6	22
13	An Empirical Investigation on Effort Estimation in Agile Global Software Development. , 2015, , .		21
14	Interactive Learning. , 2017, , .		21
15	"I know it when I see it" Perceptions of Code Quality. , 2018, , .		21
16	Developing and using checklists to improve software effort estimation: A multi-case study. Journal of Systems and Software, 2018, 146, 286-309.	3.3	20
17	The PLUSS toolkit?. , 2005, , .		17
18	On the Quality of Examples in Introductory Java Textbooks. ACM Transactions on Computing Education, 2011, 11, 1-21.	2.9	17

#	Article	IF	Citations
19	The students conference---a tool for the teaching of research, writing, and presentation skills. , 1998, , .		16
20	Evaluating and strategizing the onboarding of software developers in large-scale globally distributed projects. Journal of Systems and Software, 2020, 169, 110699.	3.3	16
21	Improving CRC-card role-play with role-play diagrams. , 2005, , .		15
22	Developing a common format for sharing programming assignments. SIGCSE Bulletin, 2008, 40, 167-182.	0.1	15
23	Regression testing for large-scale embedded software development â€“ Exploring the state of practice. Information and Software Technology, 2020, 120, 106254.	3.0	15
24	Understanding the order of agile practice introduction: Comparing agile maturity models and practitionersâ€™ experience. Journal of Systems and Software, 2019, 156, 1-20.	3.3	14
25	An evaluation of object oriented example programs in introductory programming textbooks. SIGCSE Bulletin, 2010, 41, 126-143.	0.1	13
26	Software product line modeling made practical. Communications of the ACM, 2006, 49, 49-54.	3.3	12
27	An Evolutionary Perspective on Socio-Technical Congruence: The Rubber Band Effect. , 2013, , .		12
28	Beauty and the Beast: on the readability of object-oriented example programs. Software Quality Journal, 2016, 24, 231-246.	1.4	12
29	Use Cases for Systems Engineeringâ€”An Approach and Empirical Evaluation. Systems Engineering, 2008, 11, 39-60.	1.6	11
30	Teaching software modeling in computing curricula. , 2012, , .		11
31	Assessing test artifact qualityâ€”A tertiary study. Information and Software Technology, 2021, 139, 106620.	3.0	10
32	An empirical study on the effectiveness of data resampling approaches for crossâ€project software defect prediction. IET Software, 2022, 16, 185-199.	1.5	10
33	Table compression for tree automata. ACM Transactions on Programming Languages and Systems, 1991, 13, 295-314.	1.7	9
34	Handover of managerial responsibilities in global software development: a case study of source code evolution and quality. Software Quality Journal, 2015, 23, 539-566.	1.4	8
35	Heterogeneous Systems Testing Techniques: An Exploratory Survey. Lecture Notes in Business Information Processing, 2015, , 67-85.	0.8	8
36	Test-Case Quality â€“ Understanding Practitionersâ€™ Perspectives. Lecture Notes in Computer Science, 2019, , 37-52.	1.0	8

#	Article	IF	Citations
37	Evaluating OO example programs for CS1. , 2008, , .		7
38	Team Projects in Computing Education. ACM Transactions on Computing Education, 2015, 15, 1-5.	2.9	7
39	Revisiting the Impact of Concept Drift on Just-in-Time Quality Assurance. , 2020, , .		7
40	Team Projects in Computing Education. ACM Transactions on Computing Education, 2016, 16, 1-4.	2.9	6
41	SWVP - A Requirements Prioritization Technique for Global Software Development. , 2019, , .		6
42	Experience with Work-Product Oriented Software Development Projects. Computer Science Education, 2001, 11, 111-133.	2.7	5
43	Performance Evolution of Newcomers in Large-Scale Distributed Software Projects: An Industrial Case Study. , 2019, , .		5
44	Editorial: Learning and Teaching Object Technology. Computer Science Education, 2003, 13, 243-247.	2.7	4
45	Doing your first OO project. , 1997, , .		3
46	Report on metrics 2001. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2001, 26, 52-57.	0.5	3
47	7.2.1 The FAR Approach â€“ Functional Analysis/Allocation and Requirements Flowdown Using Use Case Realizations. Incose International Symposium, 2006, 16, 950-964.	0.2	3
48	Literature review of flexibility attributes: A flexibility framework for software developing organization. Journal of Software: Evolution and Process, 2018, 30, e1937.	1.2	3
49	Pedagogies and Tools for the Teaching and Learning of Object Oriented Concepts. , 2007, , 182-192.		3
50	Information Sources and Their Importance to Prioritize Test Cases in the Heterogeneous Systems Context. Communications in Computer and Information Science, 2014, , 86-98.	0.4	3
51	Are Size Measures Better Than Expert Judgment? An Industrial Case Study on Requirements Volatility. Proceedings of the Asia Pacific Software Engineering Conference, 2007, , .	0.0	2
52	Evaluating OO example programs for CS1. SIGCSE Bulletin, 2008, 40, 47-52.	0.1	2
53	Sharing and discussing UML modeling exercises in a PLE. , 2010, , .		2
54	Ethical Considerations in Research on User Feedback. , 2017, , .		2

#	Article	IF	Citations
55	A preliminary checklist for capturing baseline situations in studying the impacts of agile practices introduction. , 2018, , .		2
56	Transitioning to OOP/Java â€" A Never Ending Story. Lecture Notes in Computer Science, 2008, , 80-97.	1.0	2
57	Towards Understanding How to Build Strategic Flexibility of an IT Organization. , 2014, , .		2
58	Tools and Environments for Learning Object-Oriented Concepts. Lecture Notes in Computer Science, 2002, , 30-43.	1.0	2
59	Worked examples for sound object-oriented pedagogy. , 2008, , .		1
60	A method for investigating the quality of evolving objectâ€oriented software using defects in global software development projects. Journal of Software: Evolution and Process, 2016, 28, 622-641.	1.2	1
61	Supporting refactoring of BDD specificationsâ€"An empirical study. Information and Software Technology, 2022, 141, 106717.	3.0	1
62	Collaborative Learning of UML and SysML. International Journal of Engineering Pedagogy, 2011, 1, 6.	0.7	1
63	Integrated software reuse: Management and techniques. Information and Software Technology, 1995, 37, 255.	3.0	0
64	Process in oo pedagogy. , 2007, , .		0
65	Good examples for exposing bad practice. , 2009, , .		0
66	The eighth "killer examples" workshop. , 2009, , .		0
67	embed4Auto. , 2010, , .		0
68	Using Role-Play Diagrams to Improve Scenario Role-Play. Lecture Notes in Computer Science, 2010, , 309-334.	1.0	0
69	What's bothering developers in code review?. , 2022, , .		0