Serge Demeyer

List of Publications by Year
in descending order

76 papers	1,417 citations	759055 12 h-index	526166 27 g-index
77 all docs	77 docs citations	77 times ranked	879 citing authors

#	Article	IF	Citations
1	Exploring actionable visualizations for environmental data: Air quality assessment of two Belgian locations. Environmental Modelling and Software, 2022, 147, 105230.	1.9	8
2	Small-Amp: Test amplification in a dynamically typed language. Empirical Software Engineering, 2022, 27, .	3.0	4
3	A comparative study of test code clones and production code clones. Journal of Systems and Software, 2021, 176, 110940.	3.3	5
4	Comparing mutation coverage against branch coverage in an industrial setting. International Journal on Software Tools for Technology Transfer, 2020, 22, 365-388.	1.7	6
5	Clone Detection in Test Code: An Empirical Evaluation. , 2020, , .		6
6	Semi-automatic Test Case Expansion for Mutation Testing. , 2020, , .		3
7	Mutant Density. , 2020, , .		0
8	Automating Software Re-engineering. Lecture Notes in Computer Science, 2020, , 3-8.	1.0	1
9	Formal Verification of Developer Tests: A Research Agenda Inspired by Mutation Testing. Lecture Notes in Computer Science, 2020, , 9-24.	1.0	3
10	An Empirical Study on Accidental Cross-Project Code Clones. , 2020, , .		1
11	Value-based technical debt management: an exploratory case study in start-ups and scale-ups. , 2019, , .		3
12	A Novel Approach for Detecting Type-IV Clones in Test Code. , 2019, , .		4
13	Do Null-Type Mutation Operators Help Prevent Null-Type Faults?. Lecture Notes in Computer Science, 2019, , 419-434.	1.0	5
14	Changes as First-Class Citizens. ACM Computing Surveys, 2018, 50, 1-38.	16.1	12
15	Evaluating the efficiency of continuous testing during test-driven development. , 2018, , .		3
16	An exploratory qualitative and quantitative analysis of emotions in issue report comments of open source systems. Empirical Software Engineering, 2018, 23, 521-564.	3.0	42
17	Goal-oriented mutation testing with focal methods. , 2018, , .		5
18	Comparing Spectrum Based Fault Localisation Against Test-to-Code Traceability Links. , 2018, , .		1

#	Article	IF	Citations
19	Test behaviour detection as a test refactoring safety. , 2018, , .		9
20	Unit tests and component tests do make a difference on fault localisation effectiveness. , 2018, , .		2
21	Speeding up mutation testing via the cloud. , 2018, , .		5
22	C++11/14 Mutation Operators Based on Common Fault Patterns. Lecture Notes in Computer Science, 2018, , 102-118.	1.0	5
23	Migrating towards microservices: migration and architecture smells. , 2018, , .		43
24	On the use of sequence mining within spectrum based fault localisation. , 2018, , .		6
25	An empirical study of clone density evolution and developer cloning tendency. , 2017, , .		2
26	DEVS for AUTOSAR-based system deployment modeling and simulation. Simulation, 2017, 93, 489-513.	1.1	11
27	Dynamic mutant subsumption analysis using LittleDarwin. , 2017, , .		3
28	Indoor environmental quality index for conservation environments: The importance of including particulate matter. Building and Environment, 2017, 126, 132-146.	3.0	23
29	On the Differences between Unit and Integration Testing in the TravisTorrent Dataset. , 2017, , .		7
30	LittleDarwin: A Feature-Rich and Extensible Mutation Testing Framework for Large and Complex Java Systems. Lecture Notes in Computer Science, 2017, , 148-163.	1.0	10
31	A game of refactoring. , 2016, , .		5
32	A Model to Estimate First-Order Mutation Coverage from Higher-Order Mutation Coverage. , 2016, , .		7
33	Estimating Story Points from Issue Reports. , 2016, , .		30
34	Fine-tuning spectrum based fault localisation with frequent method item sets. , 2016, , .		28
35	Among the Machines. , 2016, , .		47
36	Evaluating random mutant selection at class-level in projects with non-adequate test suites. , 2016, , .		8

#	Article	IF	Citations
37	Change-based test selection: an empirical evaluation. Empirical Software Engineering, 2016, 21, 1990-2032.	3.0	20
38	Circumventing refactoring masking using fine-grained change recording. , 2015, , .		7
39	Localising faults in test execution traces. , 2015, , .		8
40	Mutation testing as a safety net for test code refactoring. , 2015, , .		6
41	On the influence of maintenance activity types on the issue resolution time. , 2014, , .		21
42	A transformation-based approach to context-aware modelling. Software and Systems Modeling, 2014, 13, 191-208.	2.2	15
43	Considering Polymorphism in Change-Based Test Suite Reduction. Lecture Notes in Business Information Processing, 2014, , 166-181.	0.8	10
44	Change-Based Test Selection in the Presence of Developer Tests. , 2013, , .		18
45	Predicting Reassignments of Bug Reports - An Exploratory Investigation. , 2013, , .		12
46	An Initial Investigation into Change-Based Reconstruction of Floss-Refactorings. , 2013, , .		9
47	An Initial Investigation of a Multi-layered Approach for Optimizing and Parallelizing Real-Time Media and Audio Applications. , 2013, , .		0
48	ChEOPSJ: Change-Based Test Optimization. , 2012, , .		14
49	Preserving Aspects via Automation: A Maintainability Study. , 2011, , .		1
50	Studying the co-evolution of production and test code in open source and industrial developer test processes through repository mining. Empirical Software Engineering, 2011, 16, 325-364.	3.0	122
51	Avoiding bugs pro-actively by change-oriented programming. , 2010, , .		3
52	Reverse Engineering on the Mainframe: Lessons Learned from "In Vivo" Research. IEEE Software, 2010, 27, 30-36.	2.1	14
53	Supporting inconsistency resolution through predictive change impact analysis. , 2009, , .		2
54	Using aspect orientation in legacy environments for reverse engineering using dynamic analysisâ€"An industrial experience report. Journal of Systems and Software, 2009, 82, 668-684.	3.3	6

#	Article	IF	Citations
55	Guest Editor Introduction. Computer Languages, Systems and Structures, 2009, 35, 1.	1.4	0
56	SERIOUS: Software Evolution, Refactoring, Improvement of OperationalÂÂand Usable Systems. , 2009, , .		3
57	Establishing Traceability Links between Unit Test Cases and Units under Test. , 2009, , .		81
58	Feature location in COBOL mainframe systems: An experience report. , 2009, , .		7
59	Automatic identification of key classes in a software system using webmining techniques. Journal of Software: Evolution and Process, 2008, 20, 387-417.	1.1	65
60	Software Evolution. , 2008, , .		97
61	Mining Software Repositories to Study Co-Evolution of Production &amp; Test Code. , 2008, , .		106
62	Estimation of Test Code Changes Using Historical Release Data. , 2008, , .		9
63	Exploring the composition of unit test suites. , 2008, , .		11
64	JExample: Exploiting Dependencies between Tests to Improve Defect Localization. Lecture Notes in Business Information Processing, 2008, , 73-82.	0.8	5
65	Object-Oriented Reengineering. , 2008, , 142-153.		0
66	On The Detection of Test Smells: A Metrics-Based Approach for General Fixture and Eager Test. IEEE Transactions on Software Engineering, 2007, 33, 800-817.	4.3	98
67	Studying Versioning Information to Understand Inheritance Hierarchy Changes. , 2007, , .		7
68	Optimizing data structures at the modeling level in embedded multimedia. Journal of Systems Architecture, 2007, 53, 539-549.	2.5	4
69	A Qualitative Investigation of UML Modeling Conventions. Lecture Notes in Computer Science, 2007, , 91-100.	1.0	2
70	Characterizing the Relative Significance of a Test Smell. , 2006, , .		19
71	An Experimental Investigation of UML Modeling Conventions. Lecture Notes in Computer Science, 2006, , 27-41.	1.0	23
72	Formalizing refactorings with graph transformations. Journal of Software: Evolution and Process, 2005, 17, 247-276.	1.1	79

#	Article	IF	Citations
73	Refactoring: Current Research and Future Trends. Electronic Notes in Theoretical Computer Science, 2003, 82, 483-499.	0.9	34
74	Towards Automating Source-Consistent UML Refactorings. Lecture Notes in Computer Science, 2003, , 144-158.	1.0	43
75	Formalising Behaviour Preserving Program Transformations. Lecture Notes in Computer Science, 2002, , 286-301.	1.0	62
76	Change Impact Analysis for UML Model Maintenance. , 0, , 32-56.		4