Jesus M Gonzalez-Barahona

List of Publications by Year
in descending order

108	1,995	361413	434195
papers	citations	20 h-index	31 g-index

113	113	113	960
all docs	docs citations	times ranked	citing authors

#	Article	IF	Citations
1	On the Inequality of Contributions to Wikipedia. , 2008, , .		107
2	Macro-level software evolution: a case study of a large software compilation. Empirical Software Engineering, 2009, 14, 262-285.	3.9	79
3	Beyond source code: The importance of other artifacts in software development (a case study). Journal of Systems and Software, 2006, 79, 1233-1248.	4.5	75
4	On the reproducibility of empirical software engineering studies based on data retrieved from development repositories. Empirical Software Engineering, 2012, 17, 75-89.	3.9	75
5	Evolution and Growth in Large Libre Software Projects. , 0, , .		59
6	Reproducibility and credibility in empirical software engineering: A case study based on a systematic literature review of the use of the SZZ algorithm. Information and Software Technology, 2018, 99, 164-176.	4.4	57
7	Towards a simplification of the bug report form in eclipse. , 2008, , .		54
8	Evolution of the core team of developers in libre software projects. , 2009, , .		54
9	Towards a Theoretical Model for Software Growth. , 2007, , .		49
10	An Empirical Analysis of Technical Lag in npm Package Dependencies. Lecture Notes in Computer Science, 2018, , 95-110.	1.3	46
11	FLOSSMetrics: Free/Libre/Open Source Software Metrics. , 2009, , .		45
12	Tools for the Study of the Usual Data Sources found in Libre Software Projects. International Journal of Open Source Software and Processes, 2009, 1, 24-45.	0.6	45
13	The evolution of the laws of software evolution. ACM Computing Surveys, 2013, 46, 1-28.	23.0	43
14	Applying Social Network Analysis Techniques to Community-Driven Libre Software Projects. International Journal of Information Technology and Web Engineering, 2006, 1, 27-48.	1.6	43
15	Estimating development effort in Free/Open source software projects by mining software repositories: a case study of OpenStack. , 2014, , .		41
16	A Comprehensive Study of Software Forks: Dates, Reasons and Outcomes. International Federation for Information Processing, 2012, , 1-14.	0.4	39
17	Comparison between SLOCs and number of files as size metrics for software evolution analysis. , 2006, , .		38
18	Mining large software compilations over time. , 2006, , .		37

#	Article	IF	Citations
19	Quantitative analysis of thewikipedia community of users. , 2007, , .		37
20	On the Relation between Outdated Docker Containers, Severity Vulnerabilities, and Bugs. , 2019, , .		37
21	Lessons learned from applying social network analysis on an industrial Free/Libre/Open Source Software ecosystem. Journal of Internet Services and Applications, 2015, 6, .	2.1	36
22	How bugs are born: a model to identify how bugs are introduced in software components. Empirical Software Engineering, 2020, 25, 1294-1340.	3.9	33
23	Developer identification methods for integrated data from various sources. , 2005, , .		32
24	FLOSS 2013: a survey dataset about free software contributors: challenges for curating, sharing, and combining. , 2014, , .		32
25	Geographic origin of libre software developers. Information Economics and Policy, 2008, 20, 356-363.	3.5	30
26	Understanding How Companies Interact with Free Software Communities. IEEE Software, 2013, 30, 38-45.	1.8	29
27	Perceval. , 2018, , .		27
28	Using Social Network Analysis Techniques to Study Collaboration between a FLOSS Community and a Company. International Federation for Information Processing, 2008, , 171-186.	0.4	27
29	Women in Free/Libre/Open Source Software: The Situation in the 2010s. IFIP Advances in Information and Communication Technology, 2016, , 163-173.	0.7	27
30	Geographic location of developers at SourceForge. , 2006, , .		25
31	A Model to Understand the Building and Running Inter-Dependencies of Software. Reverse Engineering (WCRE), Working Conference on, 2007, , .	0.0	25
32	A Dataset of Scratch Programs: Scraped, Shaped and Scored. , 2017, , .		25
33	Technical Lag in Software Compilations: Measuring How Outdated a Software Deployment Is. IFIP Advances in Information and Communication Technology, 2017, , 182-192.	0.7	25
34	On the prediction of the evolution of libre software projects. Conference on Software Maintenance, Proceedings of the, 2007, , .	0.0	24
35	Forecasting the Number of Changes in Eclipse Using Time Series Analysis. , 2007, , .		24
36	Studying the laws of software evolution in a longâ€lived FLOSS project. Journal of Software: Evolution and Process, 2014, 26, 589-612.	1.6	22

#	Article	IF	Citations
37	On the Impact of Outdated and Vulnerable Javascript Packages in Docker Images. , 2019, , .		22
38	GrimoireLab: A toolset for software development analytics. PeerJ Computer Science, 2021, 7, e601.	4.5	21
39	Adapting the "staged model for software evolution" to free/libre/open source software. , 2007, , .		19
40	A formal framework for measuring technical lag in component repositories â€" and its application to npm. Journal of Software: Evolution and Process, 2019, 31, e2157.	1.6	19
41	On the Diversity of Software Package Popularity Metrics: An Empirical Study of npm. , 2019, , .		18
42	Corporate Involvement of Libre Software: Study of Presence in Debian Code over Time. International Federation for Information Processing, 2007, , 121-132.	0.4	17
43	Using Software Archaeology to Measure Knowledge Loss in Software Projects Due to Developer Turnover. , 2009, , .		16
44	What if a bug has a different origin?. , 2018, , .		16
45	Using Metrics to Track Code Review Performance. , 2017, , .		14
46	Determinism and evolution. , 2008, , .		13
47	A multi-dimensional analysis of technical lag in Debian-based Docker images. Empirical Software Engineering, 2021, 26, 1.	3.9	13
48	The MetricsGrimoire Database Collection. , 2015, , .		12
49	An Empirical Study of the Reuse of Software Licensed under the GNU General Public License. IFIP Advances in Information and Communication Technology, 2009, , 185-198.	0.7	12
50	Are Developers Fixing Their Own Bugs?. International Journal of Open Source Software and Processes, 2011, 3, 23-42.	0.6	12
51	Towards Automated Quality Models for Software Development Communities: The QualOSS and FLOSSMetrics Case. , 2010, , .		10
52	Intensive metrics for the study of the evolution of open source projects: Case studies from Apache Software Foundation projects. , 2013, , .		10
53	Trends in Free, Libre, Open Source Software Communities: From Volunteers to Companies / Aktuelle Trends in Free-, Libre-, und Open-Source-Software-Gemeinschaften: Von Freiwilligen zu Unternehmen. IT - Information Technology, 2013, 55, 173-180.	0.9	10
54	Mining student repositories to gain learning analytics. An experience report. , 2013, , .		9

#	Article	IF	Citations
55	Repositories with Public Data about Software Development. International Journal of Open Source Software and Processes, 2010, 2, 1-13.	0.6	9
56	CodeCity: On-Screen or in Virtual Reality?. , 2021, , .		9
57	Executable source code and non-executable source code: analysis and relationships. , 0, , .		8
58	A synchronous on-line competition software to improve and motivate learning. , 2012, , .		8
59	Impact of the Creation of the Mozilla Foundation in the Activity of Developers. , 2007, , .		6
60	Software Development Analytics for Xen: Why and How. IEEE Software, 2019, 36, 28-32.	1.8	6
61	A Brief History of Free, Open Source Software and Its Communities. Computer, 2021, 54, 75-79.	1.1	6
62	Volunteers in Large Libre Software Projects. , 2007, , 1-24.		6
63	Implementing a New Low-Level Tasking Support for the GNAT Runtime System?. Lecture Notes in Computer Science, 1999, , 298-307.	1.3	6
64	BugTracking: A Tool to Assist in the Identification of Bug Reports. IFIP Advances in Information and Communication Technology, 2016, , 192-198.	0.7	6
65	Collecting data about FLOSS development. , 2010, , .		5
66	Determining the Geographical distribution of a Community by means of a Time-zone Analysis. , 2016, , .		5
67	FLOSS Communities: Analyzing Evolvability and Robustness from an Industrial Perspective. International Federation for Information Processing, 2010, , 336-341.	0.4	5
68	Do More Experienced Developers Introduce Fewer Bugs?. International Federation for Information Processing, 2012, , 268-273.	0.4	5
69	Programming distributed fault tolerant systems. , 1997, , .		4
70	Real-time programming with GNAT. , 1999, , .		4
71	Managing Libre Software Distributions under a Product Line Approach. , 2008, , .		4
72	A quantitative approach to the use of the Wikipedia. , 2009, , .		4

#	Article	IF	Citations
73	On the Analysis of Contributions from Privileged Users in Virtual Open Communities. , 2009, , .		4
74	Characterization of the Xen project code review process. , 2016, , .		4
75	[Engineering Paper] Graal: The Quest for Source Code Knowledge. , 2018, , .		4
76	Towards automated, provenance-driven security audit for git-based repositories: applied to germany's corona-warn-app: vision paper. , 2020, , .		4
77	Research friendly software repositories. , 2009, , .		3
78	Free/Open Source Software projects as early MOOCs. , 2014, , .		3
79	Provenance-Based Security Audits and Its Application to COVID-19 Contact Tracing Apps. Lecture Notes in Computer Science, 2021, , 88-105.	1.3	3
80	A Preliminary Analysis of Localization in Free Software: How Translations Are Performed. IFIP Advances in Information and Communication Technology, 2013, , 153-167.	0.7	3
81	Tools and Datasets for Mining Libre Software Repositories. , 0, , 24-42.		3
82	Assessing FLOSS Communities: An Experience Report from the QualOSS Project. IFIP Advances in Information and Communication Technology, 2009, , 364-364.	0.7	3
83	Software Development Metrics With a Purpose. Computer, 2022, 55, 66-73.	1.1	3
84	How Much Time Did It Take to Notify a Bug? Two Case Studies: ElasticSearch and Nova. , 2017, , .		2
85	Las ventajas de la apertura. Profesional De La Informacion, 2008, 17, 5-7.	2.7	2
86	Applying Social Network Analysis Techniques to Community-Driven Libre Software Projects. , 2009, , 28-50.		2
87	BabiaXR: Virtual Reality software data visualizations for the Web. , 2022, , .		2
88	Towards the improvement of the software quality: An Enterprise 2.0 architecture for distributed software developments. , 2008, , .		1
89	Hybrid educational worlds. , 2012, , .		1
90	A Service Based Development Environment on Web 2.0 Platforms. Lecture Notes in Computer Science, 2008, , 38-48.	1.3	1

#	Article	IF	Citations
91	Code Review Analytics: WebKit as Case Study. IFIP Advances in Information and Communication Technology, 2014, , 1-10.	0.7	1
92	Quantitative Analysis of the Top Ten Wikipedias. Communications in Computer and Information Science, 2008, , 257-268.	0.5	1
93	The Networked Forge: New Environments for Libre Software Development. International Federation for Information Processing, 2008, , 299-306.	0.4	1
94	Characterizing outdateness with technical lag. , 2020, , .		1
95	Building modular communication systems in Ada: The Simple-Com approach. Lecture Notes in Computer Science, 1998, , 225-237.	1.3	0
96	Towards predictor models for large libre software projects. , 2005, , .		0
97	Edukalibre: a tool for collaborative creation of educational material. International Journal of Continuing Engineering Education and Life-Long Learning, 2007, 17, 15.	0.2	0
98	Evaluation of FLOSS by Analyzing Its Software Evolution. Journal of Information Technology Research, 2015, 8, 62-81.	0.5	0
99	Message from MSR 2017 General Chairs. , 2017, , .		0
100	Hosting of Libre Software Projects: A Distributed Peer-to-Peer Approach. Lecture Notes in Computer Science, 2003, , 207-211.	1.3	0
101	Towards Community-Driven Development of Educational Materials: The Edukalibre Approach. Lecture Notes in Computer Science, 2006, , 125-139.	1.3	0
102	Quantitative analysis and characterization of Wikipedia requests. , 2008, , .		0
103	4th International Workshop on Public Data about Software Development. IFIP Advances in Information and Communication Technology, 2009, , 351-352.	0.7	0
104	WoPDaSD 2010: 5th Workshop on Public Data about Software Development. International Federation for Information Processing, 2010, , 421-422.	0.4	0
105	FLOSS Education: Long-Term Sustainability. International Federation for Information Processing, 2012, , 400-400.	0.4	0
106	Are Developers Fixing Their Own Bugs?. , 0, , 79-98.		0
107	Tools and Datasets for Mining Libre Software Repositories. , 0, , 564-582.		0
108	How to Gather FLOSS Metrics. , 0, , 361-362.		0