Lech Madeyski — lista publikacji i wybranych cytowań

[1] Marian Jureczko and Lech Madeyski. Cross-project defect prediction: an empirical study. *Cybernetics and Systems*, 2011. (submitted).

[2] Lech Madeyski and Marian Jureczko. Which process metrics improve software defect prediction models? An empirical study. *Information and Software Technology*, 2011. (submitted).

[3] Lech Madeyski. *Test-Driven Development: An Empirical Evaluation of Agile Practice*. Springer, (Heidelberg, London, New York), 2010. URL: `http://www.springer.com/978-3-642-04287-4`. Foreword by Prof. Claes Wohlin.

Cited by:

[3.1] Matjaz Pancur and Mojca Ciglaric. Impact of test-driven development on productivity, code and tests: A controlled experiment. *Information and Software Technology*, 53(6):557–573, 2011. (ISI impact factor 2009: 1.821).

[3.2] Harekrishna Misra. Review of "Test-driven development : an empirical evaluation of agile practice". *ACM Computing Reviews*, January, 2011. Review: CR138756.

[3.3] Janusz Górski and Krzysztof Wyrzykowski. A method of supporting client-provider cooperation in software acquisition processes and its evaluation criteria. *Przegląd Elektrotechniczny (Electrical Review)*, 86(9):188–192, 2010. (ISI impact factor 2009: 0.196).

[3.4] Marian Jureczko and Krzysztof Wyrzykowski. Automated acceptance testing tools for web applications using test-driven development. *Przegląd Elektrotechniczny (Electrical Review)*, 86(9):198–202, 2010. (ISI impact factor 2009: 0.196).

[3.5] Marko Ikonen. Leadership in Kanban Software Development Projects: A Quasi-controlled Experiment. In Pekka Abrahamsson and Nilay V. Oza, editors, *Lean Enterprise Software and Systems - First International Conference, LESS 2010, Helsinki, Finland, October 17-20, 2010*, volume 65 of *Lecture Notes in Business Information Processing*, pages 85–98. Springer, 2010. URL: http://dx.doi.org/10.1007/978-3-642-16416-3.

[3.6] Marko Ikonen, Elena Pirinen, Fabian Fagerholm, Petri Kettunen, and Pekka Abrahamsson. On the Impact of Kanban on Software Project Work: An Empirical Case Study Investigation. In *Proceedings of the 16th IEEE*

International Conference on Engineering of Complex Computer Systems (ICECCS 2011), pages 305–314. IEEE Computer Society, 2011.

[4] Lech Madeyski and Norbert Radyk. Judy – A Mutation Testing Tool for Java. *IET Software Journal (formerly IEE Proceedings Software)*, 4(1):32–42, IET, 2010. URL: `http://dx.doi.org/doi:10.1049/iet-sen.2008.0038`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski10b.pdf (impact factor 2009: 0.650; MNiSW: 13).

Cited by:

[4.1] Matjaz Pancur and Mojca Ciglaric. Impact of test-driven development on productivity, code and tests: A controlled experiment. *Information and Software Technology*, 53(6):557–573, 2011. (ISI impact factor 2009: 1.821).

[4.2] Reza Meimandi Parizi and Abdul Azim Abdul Ghani. Towards automated monitoring and forecasting of probabilistic quality properties in open source software (oss): A striking hybrid approach. In *ACIS International Conference on Software Engineering Research, Management and Applications*, pages 329–334, Los Alamitos, CA, USA, 2010. IEEE Computer Society.

[4.3] Juan Boubeta Puig, Inmaculada Medina Bulo, and Antonio García Domínguez. Equivalencias entre los operadores de mutación definidos para WS-BPEL 2.0 y los definidos para otros lenguajesb. *Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos*, 4(5):1–8, 2010.

[4.4] Juan Boubeta Puig. Implementación de operadores de mutación para WS-BPEL 2.0. Master's thesis, Universidad de Cadiz, 2010.

[4.5] Inmaculada Medina Bulo, Lorena Gutiérrez Madronal, and Juan José Domínguez Jiménez. Propuesta de optimización en la prueba de mutaciones en Java. *Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos*, 4(5):25–31, 2010.

[4.6] Stefan Buchholz. Mutation Tool for Eiffel Code Transformation. Master's thesis, ETH Zurich, 2010.

[4.7] Kapil Kumar, P. K. Gupta, and Roshan Parjapat. New Mutants Generation for Testing Java Programs. In Vinu V Das, Janahanlal Stephen, and Yogesh Chaba, editors, *Computer Networks and Information Technologies*, volume 142 of *Communications in Computer and Information Science*, pages 290–294. Springer Berlin Heidelberg, 2011.

[5] Lech Madeyski. The impact of test-first programming on branch coverage and mutation score indicator of unit tests: An experiment. *Information and Software Technology*, 52(2):169–184, Elsevier, 2010. URL: `http://dx.doi.org/10.1016/j.infsof.2009.08.007`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski10c.pdf (impact factor 2009: 1.821; MNiSW: 27).

Cited by:

[5.1] Matjaz Pancur and Mojca Ciglaric. Impact of test-driven development on productivity, code and tests: A controlled experiment. *Information and Software Technology*, 53(6):557–573, 2011. (ISI impact factor 2009: 1.821).

[5.2] Sami Kollanus. Test-driven development - still a promising approach? In *Proceedings of the International Conference on the Quality of Information and Communications Technology*, pages 403–408, Los Alamitos, CA, USA, 2010. IEEE Computer Society.

[5.3] Andre Abe Vicente and Marcio Eduardo Delamaro. ATMM uma ferramenta para gerenciamento de metricas de teste no contexto de metodos ageis. In *Workshop Brasileiro de Metodos Ageis*, pages 15–28, 2010.

[5.4] Adnan Causevic, Daniel Sundmark, and Sasikumar Punnekkat. Factors Limiting Industrial Adoption of Test Driven Development: A Systematic Review. In *Software Testing, Verification and Validation (ICST), 2011 IEEE Fourth International Conference on*, pages 337 –346, March 2011.

[5.5] Sami Kollanus. Critical issues on test-driven development. In Danilo Caivano, Markku Oivo, Maria Teresa Baldassarre, and Giuseppe Visaggio, editors, *Proceedings of the Product-Focused Software Process Improvement - 12th International Conference, PROFES 2011, Torre Canne, Italy, June 20-22, 2011.*, volume 6759 of *Lecture Notes in Business Information Processing*, pages 322–336, Berlin, Heidelberg, 2011. Springer.

[5.6] Ilona Bluemke and Karol Kulesza. A Comparison of Dataflow and Mutation Testing of Java Methods. In Wojciech Zamojski, Janusz Kacprzyk, Jacek Mazurkiewicz, Jaroslaw Sugier, and Tomasz Walkowiak, editors, *Dependable Computer Systems*, volume 97 of *Advances in Intelligent and Soft Computing*, pages 17–30. Springer Berlin / Heidelberg, 2011. 10.1007/978-3-642-21393-9.

[6] Lech Madeyski. Empirical studies on the impact of test-first programming. Technical Report I32/10/P-004, Wroclaw University of Technology, Institute of Informatics, 2010. http://madeyski.e-informatyka.pl/download/Madeyski09TFStudies.pdf.

[7] Marian Jureczko and Lech Madeyski. Predykcja defektów na podstawie metryk oprogramowania - identyfikacja klas projektów. In Janusz Górski and Cezary Orłowski, editors, *Inżynieria Oprogramowania w procesach integracji systemów informatycznych*, pages 185–192. Wydawnictwo Komunikacji i Łączności, 2010. Draft: http://madeyski.e-informatyka.pl/download/JureczkoMadeyski10e.pdf (MNiSW: 3).

[8] Marian Jureczko and Lech Madeyski. Towards identifying software project clusters with regard to defect prediction. In *Proceedings of PROMISE'2010: 6th International Conference on Predictive Models in Software Engineering*. ACM Press, 2010. URL: `http:`

//dx.doi.org/10.1145/1868328.1868342. Draft: http://madeyski.e-informatyka.pl/download/JureczkoMadeyski10f.pdf (MNiSW: ?).

Cited by:

[8.1] Zhimin He, Fengdi Shu, Ye Yang, Mingshu Li, and Qing Wang. An investigation on the feasibility of cross-project defect prediction. *Automated Software Engineering*, pages 1–33. 10.1007/s10515-011-0090-3, (ISI impact factor 2009: 1.267).

[8.2] Marian Jureczko. Significance of different software metrics in defect prediction. *Software engineering : an international Journal*, 1:86–95, 2011.

[9] Zbigniew Huzar and Lech Madeyski. Editorial. *e-Informatica Software Engineering Journal*, 4(1):7–8, 2010. URL: http://www.e-informatyka.pl/attach/e-Informatica_-_Volume_4/eInformatica2010Editorial.pdf.

[10] Zbigniew Huzar and Lech Madeyski. Editorial. *e-Informatica Software Engineering Journal*, 3(1):7–7, 2009. URL: http://www.e-informatyka.pl/attach/e-Informatica_-_Volume_3/eInformatica2009Editorial.pdf.

[11] Lech Madeyski. The Impact of Pair Programming on Thoroughness and Fault Detection Effectiveness of Unit Tests Suites. *Wiley, Software Process: Improvement and Practice*, 13(3):281–295, Wiley, 2008. URL: http://dx.doi.org/10.1002/spip.382. Draft: http://madeyski.e-informatyka.pl/download/Madeyski08.pdf (MNiSW: 2).

Cited by:

[11.1] Adnan Causevic, Daniel Sundmark, and Sasikumar Punnekkat. Factors Limiting Industrial Adoption of Test Driven Development: A Systematic Review. In *Software Testing, Verification and Validation (ICST), 2011 IEEE Fourth International Conference on*, pages 337 –346, March 2011.

[12] Tomas Hruska, Lech Madeyski, and Miroslaw Ochodek, editors. *Software Engineering Techniques in Progress, 3rd IFIP Central and East European Conference on Software Engineering Techniques, CEE-SET 2008, Poznań, Poland*. Wrocław University of Technology, 2008.

[13] Zbigniew Huzar and Lech Madeyski. Editorial. *e-Informatica Software Engineering Journal*, 2(1):7–7, 2008. URL: http://www.e-informatyka.pl/attach/e-Informatica_-_Volume_2/Vol2Iss1EditorialeInformatica.pdf.

[14] Lech Madeyski and Wojciech Biela. Capable Leader and Skilled and Motivated Team Practices to Introduce eXtreme Programming. *Lecture Notes in Computer Science*, 5082:96–102, Springer, 2008. URL: `http://dx.doi.org/10.1007/978-3-540-85279-7_8`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski08d.pdf (MNiSW: 13).

Cited by:

[14.1] Ani Liza Asnawi, Andrew M. Gravell, and Gary B. Wills. Empirical investigation on agile methods usage: Issues identified from early adopters in malaysia. In *XP'11: Agile Processes in Software Engineering and Extreme Programming - 12th International Conference, XP 2011, Madrid, Spain, May 10-13, 2011*, pages 192–207. Springer, 2011.

[14.2] Ani Liza Asnawi, Andrew M Gravell, and Gary B Wills. An empirical study: Understanding factors and barriers for implementing agile methods in malaysia. In *Proceedings of the 5th International Doctoral Symposium on Empirical Software Engineering (IDoESE)*, 2010.

[15] Lech Madeyski. On the Effects of Pair Programming on Thoroughness and Fault-Finding Effectiveness of Unit Tests. *Lecture Notes in Computer Science*, 4589:207–221, Springer, 2007. URL: `http://dx.doi.org/10.1007/978-3-540-73460-4_20`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski07.pdf, (MNiSW: 13).

Cited by:

[15.1] Norsaremah Salleh, Emilia Mendes, and John Grundy. Empirical studies of pair programming for cs/se teaching in higher education: A systematic literature review. *IEEE Transactions on Software Engineering*, 37(4):509–525, july-aug. 2011. (ISI impact factor 2009: 3.75).

[15.2] Jo E. Hannay, Tore Dybå, Erik Arisholm, and Dag I. K. Sjøberg. The effectiveness of pair programming: A meta-analysis. *Information and Software Technology*, 51(7):1110–1122, 2009. (ISI impact factor 2009: 1.821).

[15.3] VenuGopal Balijepally, RadhaKanta Mahapatra, Sridhar Nerur, and Kenneth H. Price. Are Two Heads Better than One for Software Development? The Productivity Paradox of Pair Programming. *MIS Quarterly*, 33(1):91–118, March 2009. (ISI impact factor 2008: 5.183).

[15.4] Anna Szustek Anna Derezińska. Object-Oriented Testing Capabilities and Performance Evaluation of the C# Mutation System. In *CEE-SET'09: Balancing Agility and Formalism in Software Engineering, Second IFIP TC 2 Central and East European Conference on Software Engineering Techniques, CEE-SET 2009, Kraków, Poland, October 12-14, 2009, Revised Selected Papers*. Springer, 2010.

[15.5] Adnan Causevic, Daniel Sundmark, and Sasikumar Punnekkat. Factors Limiting Industrial Adoption of Test Driven Development: A Systematic

Review. In *Software Testing, Verification and Validation (ICST), 2011 IEEE Fourth International Conference on*, pages 337 –346, March 2011.

[15.6] Marian Jureczko. The level of agility in testing process in a large scale financial software project. In Tomas Hruska, Lech Madeyski, and Miroslaw Ochodek, editors, *Software Engineering Techniques in Progress*. Wroclaw University of Technology Press, 2008.

[16] Lech Madeyski and Wojciech Biela. Empirical Evidence Principle and Joint Engagement Practice to Introduce XP. *Lecture Notes in Computer Science*, 4536:141–144, Springer, 2007. URL: `http://dx.doi.org/10.1007/978-3-540-73101-6_19`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski07b.pdf (MNiSW: 13).

[17] Zbigniew Huzar and Lech Madeyski. Empirical Evaluation of Novel Approaches to Software Engineering. *e-Informatica Software Engineering Journal*, 1(1):7–8, 2007. URL: `http://www.e-informatyka.pl/attach/e-Informatica_-_Volume_1/Vol1Iss1EditorialeInformatica.pdf`.

[18] Lech Madeyski and Łukasz Szała. The Impact of Test-Driven Development on Software Development Productivity – An Empirical Study. *Lecture Notes in Computer Science*, 4764:200–211, Springer, 2007. URL: `http://dx.doi.org/10.1007/978-3-540-75381-0_18`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski07d.pdf (MNiSW: 13).

Cited by:

[18.1] Albert Tort, Antoni Olivé, and Maria-Ribera Sancho. An approach to test-driven development of conceptual schemas. *Data & Knowledge Engineering*, 70(12):1088 – 1111, 2011. (ISI impact factor 2010: 1.717).

[18.2] Tomaz Dogsa and David Batic. The effectiveness of test-driven development: an industrial case study. *Software Quality Journal*, 19:643–661, 2011. 10.1007/s11219-011-9130-2, (ISI impact factor 2010: 0.75).

[18.3] Roy Oberhauser. Towards Automated Test Practice Detection and Governance. In Alimohammad, A and Meixner, A and Popescu, M, editor, *2009 First International Conference on Advances in System Testing and Validation Lifecycle, Oporto, Portugal, September 20-25, 2009*, pages 19–24. IEEE, 2009.

[18.4] Hongbing Kou, Philip M. Johnson, and Hakan Erdogmus. Operational definition and automated inference of test-driven development with Zorro. *Automated Software Engineering*, 17(1):57–85, March 2010. (ISI impact factor 2009: 1.267).

[18.5] Sami Kollanus. Test-driven development - still a promising approach? In *Proceedings of the International Conference on the Quality of Information*

and Communications Technology, pages 403–408, Los Alamitos, CA, USA, 2010. IEEE Computer Society.

[18.6] Andy Oram and Greg Wilson, editors. *Making Software: What Really Works, and Why We Believe It*. O'Reilly Media, 1st edition, 2010.

[18.7] Adnan Causevic, Daniel Sundmark, and Sasikumar Punnekkat. Factors Limiting Industrial Adoption of Test Driven Development: A Systematic Review. In *Software Testing, Verification and Validation (ICST), 2011 IEEE Fourth International Conference on*, pages 337 –346, March 2011.

[18.8] Sami Kollanus. Critical issues on test-driven development. In Danilo Caivano, Markku Oivo, Maria Teresa Baldassarre, and Giuseppe Visaggio, editors, *Proceedings of the Product-Focused Software Process Improvement - 12th International Conference, PROFES 2011, Torre Canne, Italy, June 20-22, 2011.*, volume 6759 of *Lecture Notes in Business Information Processing*, pages 322–336, Berlin, Heidelberg, 2011. Springer.

[19] Lech Madeyski, Miroslaw Ochodek, Dawid Weiss, and Jaroslav Zendulka, editors. *Software Engineering in Progress, 2nd IFIP Central and East European Conference on Software Engineering Techniques, CEE-SET 2007, Poznań, Poland*. Nakom, 2007.

[20] Lech Madeyski and Łukasz Szała. Impact of aspect-oriented programming on software development efficiency and design quality: an empirical study. *IET Software Journal (formerly IEE Proceedings Software)*, 1(5):180–187, IET, 2007. URL: `http://dx.doi.org/10.1049/iet-sen:20060071`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski07g.pdf (impact factor 2007: 0.400; MNiSW: 13).

Cited by:

[20.1] Mohammad Alshayeb, Hamdi Al-Jamimi, and Mahmoud O. Elish. Empirical taxonomy of refactoring methods for aspect-oriented programming. *Journal of Software Maintenance and Evolution: Research and Practice*, pages n/a–n/a, 2011. (ISI impact factor 2010: 0.606).

[20.2] Stefan Endrikat and Stefan Hanenberg. Is Aspect-Oriented Programming a Rewarding Investment into Future Code Changes? A Socio-technical Study on Development and Maintenance Time. In *Program Comprehension (ICPC), 2011 IEEE 19th International Conference on*, pages 51–60, june 2011.

[20.3] Stefan Hanenberg, Sebastian Kleinschmager, and Manuel Josupeit-Walter. Does Aspect-Oriented Programming Increase the Development Speed for Crosscutting Code? An Empirical Study. In *ESEM'09: 3rd International Symposium on Empirical Software Engineering and Measurement, Lake Buena Vista, FL, October 15-16, 2009*, pages 156–167. IEEE, 2009.

[20.4] Muhammad Sarmad Ali, Muhammad Ali Babar, Lianping Chen, and Klaas-Jan Stol. A systematic review of comparative evidence of aspect-oriented programming. *Information and Software Technology*, 52(9):871 – 887, 2010. (ISI impact factor 2009: 1.821).

[20.5] Sebastian Kleinschmager Manuel Josupeit-Walter and Stefan Hanenberg. Preliminary Results of an Experiment Repetition for Measuring the Impact of Aspect-Oriented Programming on Development Time. In Stefan Hanenberg and Alessandro Garcia and Phil Greenwood and Eduardo Figueiredo, editor, *Empirical Evaluation of Software Composition Techniques (ESCOT 2010) at AOSD 2010, Saint-Malo, France 2010*, 2010.

[20.6] Felix-Antoine Bourbonnais and Luc Lamontagne. Using AOP for an Academic Agile Project: A Pilot Study. In Stefan Hanenberg and Alessandro Garcia and Phil Greenwood and Eduardo Figueiredo, editor, *Empirical Evaluation of Software Composition Techniques (ESCOT 2010) at AOSD 2010, Saint-Malo, France 2010Vic*, 2010.

[20.7] Amjed Tahir, Rodina Ahmad, and Zarinah Mohd Kasirun. Maintainability Dynamic Metrics Data Collection Based On Aspect-Oriented Technology. *Malaysian Journal of Computer Science*, 23(3):177–194, 2010. (ISI without impact factor).

[20.8] Adam Przybylek. Where the truth lies: Aop and its impact on software modularity. In Dimitra Giannakopoulou and Fernando Orejas, editors, *Fundamental Approaches to Software Engineering - 14th International Conference, FASE 2011, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2011, Saarbrücken, Germany, March 26-April 3, 2011. Proceedings*, volume 6603 of *Lecture Notes in Computer Science*, pages 447–461. Springer, 2011.

[20.9] Adam Przybylek. An empirical assessment of the impact of aspect-oriented programming on software modularity. In Pericles Loucopoulos and Leszek A. Maciaszek, editors, *ENASE 2010 - Proceedings of the Fifth International Conference on Evaluation of Novel Approaches to Software Engineering, Athens, Greece, July 22-24, 2010*, pages 139–148. SciTePress, 2010.

[21] Lech Madeyski and Zbigniew Huzar. External code quality model and cross-validation of the model. Technical Report PRE I31/07/P-003, Institute of Applied Informatics, Wroclaw University of Technology, 2007. URL: `http://madeyski.e-informatyka.pl/download/Madeyski07i.pdf`.

[22] Lech Madeyski. The Impact of Pair Programming and Test-Driven Development on Package Dependencies in Object-Oriented Design – An Experiment. *Lecture Notes in Computer Science*, 4034:278–289, Springer, 2006. URL: `http://dx.doi.org/10.1007/11767718_24`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski06.pdf; (impact factor 2005: 0.402; MNiSW: 13).

Cited by:

[22.1] Norsaremah Salleh, Emilia Mendes, and John Grundy. Empirical studies of pair programming for cs/se teaching in higher education: A systematic literature review. *IEEE Transactions on Software Engineering*, 37(4):509 –525, july-aug. 2011. (ISI impact factor 2009: 3.75).

[22.2] Matjaz Pancur and Mojca Ciglaric. Impact of test-driven development on productivity, code and tests: A controlled experiment. *Information and Software Technology*, 53(6):557–573, 2011. (ISI impact factor 2009: 1.821).

[22.3] Andy Oram and Greg Wilson, editors. *Making Software: What Really Works, and Why We Believe It*. O'Reilly Media, 1st edition, 2010.

[22.4] Kim Man Lui and Keith C. C. Chan. Software process fusion by combining pair and solo programming. *IET Software Journal (formerly IEE Proceedings Software)*, 2(4):379–390, 2008. (ISI impact factor 2008: 0.543).

[22.5] Jo E. Hannay, Tore Dybå, Erik Arisholm, and Dag I. K. Sjøberg. The effectiveness of pair programming: A meta-analysis. *Information and Software Technology*, 51(7):1110–1122, 2009. (ISI impact factor 2009: 1.821).

[22.6] VenuGopal Balijepally, RadhaKanta Mahapatra, Sridhar Nerur, and Kenneth H. Price. Are Two Heads Better than One for Software Development? The Productivity Paradox of Pair Programming. *MIS Quarterly*, 33(1):91–118, March 2009. (ISI impact factor 2008: 5.183).

[22.7] Alexander Serebrenik, Serguei A. Roubtsov, and Mark van den Brand. Dn-based architecture assessment of Java Open Source software systems. In *ICPC'09: IEEE 17th International Conference on Program Comprehension, 2009*, pages 198–207, May 2009.

[22.8] Matthias Müller. *Analyse leichtgewichtiger Softwareentwicklungsmethoden*. Logos, 2008. (ISBN: 978-3-8325-2073-1).

[22.9] Sami Kollanus. Test-driven development - still a promising approach? In *Proceedings of the International Conference on the Quality of Information and Communications Technology*, pages 403–408, Los Alamitos, CA, USA, 2010. IEEE Computer Society.

[22.10] Norsaremah Salleh. *Investigating the Effect of Students' Personality Traits Towards Improving Pair Programming's Effectiveness as a Pedagogical Tool for CS/SE Education*. PhD thesis, The University of Auckland, Auckland, New Zealand, 2010.

[22.11] Nattakarn Phaphoom, Alberto Sillitti, and Giancarlo Succi. Pair programming and software defects – an industrial case study. In Wil Aalst, John Mylopoulos, Michael Rosemann, Michael J. Shaw, Clemens Szyperski, Alberto Sillitti, Orit Hazzan, Emily Bache, and Xavier Albaladejo, editors, *Agile Processes in Software Engineering and Extreme Programming*, volume 77 of *Lecture Notes in Business Information Processing*, pages 208–222. Springer Berlin Heidelberg, 2011. 10.1007/978-3-642-20677-1.

[22.12] Adnan Causevic, Daniel Sundmark, and Sasikumar Punnekkat. Factors Limiting Industrial Adoption of Test Driven Development: A Systematic

Review. In *Software Testing, Verification and Validation (ICST), 2011 IEEE Fourth International Conference on*, pages 337 –346, March 2011.

[22.13] Sami Kollanus. Critical issues on test-driven development. In Danilo Caivano, Markku Oivo, Maria Teresa Baldassarre, and Giuseppe Visaggio, editors, *Proceedings of the Product-Focused Software Process Improvement - 12th International Conference, PROFES 2011, Torre Canne, Italy, June 20-22, 2011.*, volume 6759 of *Lecture Notes in Business Information Processing*, pages 322–336, Berlin, Heidelberg, 2011. Springer.

[23] Lech Madeyski. Is External Code Quality Correlated with Programming Experience or Feelgood Factor? *Lecture Notes in Computer Science*, 4044:65–74, Springer, 2006. URL: `http://dx.doi.org/10.1007/11774129_7`. Draft: http://madeyski.e-informatyka.pl/download/Madeyski06b.pdf (impact factor 2005: 0.402; MNiSW: 13).

Cited by:

[23.1] Norsaremah Salleh, Emilia Mendes, and John Grundy. Empirical studies of pair programming for cs/se teaching in higher education: A systematic literature review. *IEEE Transactions on Software Engineering*, 37(4):509 –525, july-aug. 2011. (ISI impact factor 2009: 3.75).

[23.2] Matthias Müller. *Analyse leichtgewichtiger Softwareentwicklungsmethoden.* Logos, 2008. (ISBN: 978-3-8325-2073-1).

[23.3] Norsaremah Salleh. *Investigating the Effect of Students' Personality Traits Towards Improving Pair Programming's Effectiveness as a Pedagogical Tool for CS/SE Education.* PhD thesis, The University of Auckland, Auckland, New Zealand, 2010.

[23.4] Panagiotis Sfetsos and Ioannis Stamelos. Empirical studies on quality in agile practices: A systematic literature review. In *Quality of Information and Communications Technology (QUATIC), 2010 Seventh International Conference on the*, pages 44 –53, 29 2010-oct. 2 2010.

[24] Lech Madeyski and Michał Stochmiałek. Architectural Design of Modern Web Applications. *Foundations of Computing and Decision Sciences Journal*, 30(1):49–60, 2005. URL: `http://madeyski.e-informatyka.pl/download/23.pdf`. Scholar Google citations ≥ 9 (MNiSW: 9).

Cited by:

[24.1] Yasmeen Ahmad, François-Michel Boisvert, Peter Gregor, Andy Cobley, and Angus I. Lamond. Nopdb: Nucleolar proteome database - 2008 update. *Nucleic Acids Research*, 37(Database-Issue):181–184, 2009. (ISI impact factor 2009: 7.479).

[24.2] Melek Oktay, Ayşe Betül Gülbağcı, and Mustafa Sarıöz. Architectural, Technological and Performance Issues in Enterprise Applications. *International Journal of Computer and Information Science and Engineering*, 1(2):114–119, 2007.

[25] Lech Madeyski. Preliminary Analysis of the Effects of Pair Programming and Test-Driven Development on the External Code Quality. In Krzysztof Zieliński and Tomasz Szmuc, editors, *Software Engineering: Evolution and Emerging Technologies*, volume 130 of *Frontiers in Artificial Intelligence and Applications*, pages 113–123. IOS Press, 2005. URL: `http://madeyski.e-informatyka.pl/download/Madeyski05b.pdf`. (MNiSW: 7).

Cited by:

[25.1] Norsaremah Salleh, Emilia Mendes, and John Grundy. Empirical studies of pair programming for cs/se teaching in higher education: A systematic literature review. *IEEE Transactions on Software Engineering*, 37(4):509 –525, july-aug. 2011. (ISI impact factor 2009: 3.75).

[25.2] Matjaz Pancur and Mojca Ciglaric. Impact of test-driven development on productivity, code and tests: A controlled experiment. *Information and Software Technology*, 53(6):557–573, 2011. (ISI impact factor 2009: 1.821).

[25.3] Tomaz Dogsa and David Batic. The effectiveness of test-driven development: an industrial case study. *Software Quality Journal*, 19:643–661, 2011. 10.1007/s11219-011-9130-2, (ISI impact factor 2010: 0.75).

[25.4] Steve Bannerman and Andrew Martin. A multiple comparative study of test-with development product changes and their effects on team speed and product quality. *Empirical Software Engineering*, 16:177–210, 2011. (ISI impact factor 2009: 1.612).

[25.5] Andy Oram and Greg Wilson, editors. *Making Software: What Really Works, and Why We Believe It*. O'Reilly Media, 1st edition, 2010.

[25.6] Ron Jeffries and Grigori Melnik. TDD–The Art of Fearless Programming. *IEEE Software*, 24(3):24–30, 2007. (ISI impact factor 2007: 1.462).

[25.7] Kim Man Lui and Keith C. C. Chan. Software process fusion by combining pair and solo programming. *IET Software Journal (formerly IEE Proceedings Software)*, 2(4):379–390, 2008. (ISI impact factor 2008: 0.543).

[25.8] Grigori Melnik. *Empirical Analyses of Executable Acceptance Test Driven Development*. PhD thesis, University of Calgary, Calgary, Alberta, Canada, 2007.

[25.9] Chetan Desai, David Janzen, and Kyle Savage. A survey of evidence for test-driven development in academia. *SIGCSE Bull.*, 40(2):97–101, 2008.

[25.10] Burak Turhan, Ayse Bener, Pasi Kuvaja, and Markku Oivo. A quantitative comparison of test-first and test-last code in an industrial project. In Alberto Sillitti, Angela Martin, Xiaofeng Wang, and Elizabeth Whitworth, editors, *Agile Processes in Software Engineering and Extreme Programming*, volume 48 of *Lecture Notes in Business Information Processing*, pages 232–237. Springer Berlin Heidelberg, 2010.

[25.11] Norsaremah Salleh. A systematic review of pair programming research - Initial Results. In *NZCSRSC'08: The New Zealand Computer Science*

Research Students Conference, Christchurch, New Zealand, April 14–18, 2008, pages 151–158, 2008.

[25.12] Vikram S Bhadauria. *To test before or to test after - an experimental investigation of the impact of Test Driven Development.* PhD thesis, The University of Texas at Arlington, Arlington, TX, USA, 2009.

[25.13] Lutz Prechelt, Ulrich Stärk, and Stephan Salinger. Types of Cooperation Episodes in Side-by-Side Programming. In *PPIG*. Psychology of Programming Interest Group, 2009. http://www.ppig.org/papers/21st-prechelt.pdf.

[25.14] Sami Kollanus. Test-driven development - still a promising approach? In *Proceedings of the International Conference on the Quality of Information and Communications Technology*, pages 403–408, Los Alamitos, CA, USA, 2010. IEEE Computer Society.

[25.15] Sami Kollanus. Critical issues on test-driven development. In Danilo Caivano, Markku Oivo, Maria Teresa Baldassarre, and Giuseppe Visaggio, editors, *Proceedings of the Product-Focused Software Process Improvement - 12th International Conference, PROFES 2011, Torre Canne, Italy, June 20-22, 2011.*, volume 6759 of *Lecture Notes in Business Information Processing*, pages 322–336, Berlin, Heidelberg, 2011. Springer.

[25.16] Kotrappa Sirbi and Prakash Jayanth Kulkarni. Impact of Aspect Oriented Programming on Software Development Quality Metrics. *Global Journal of Computer Science and Technology*, 10(7):28–36, 2010.

[25.17] Gerardo Canfora and Corrado Aaron Visaggio. Measuring the impact of testing on code structure in test driven development: metrics and empirical analysis. In Giulio Concas and Michele Marchesi, editors, *First International Symposium on Emerging Trends in Software Metrics*, pages 23–32, 2009.

[26] Lech Madeyski. An empirical analysis of the impact of pair programming and test-driven development on CK design complexity metrics. Technical Report PRE I31/05/P-004, Institute of Applied Informatics, Wroclaw University of Technology, 2005.

[27] Lech Madeyski, Adam Mrozowski, and Sebastian Gil. Komunikacja w czasie rzeczywistym w sieci Internet. In Andrzej Kwiecień i Piotr Gaj, editor, *Współczesne problemy systemów czasu rzeczywistego*, pages 375–386. Wydawnictwa Naukowo-Techniczne, 2004. URL: `http://madeyski.e-informatyka.pl/download/18.pdf`. (MNiSW: 3).

[28] Lech Madeyski and Michał Stochmiałek. Architektura Nowoczesnych Aplikacji Internetowych. In Jerzy Górski and Andrzej Wardziński, editors, *Inżynieria Oprogramowania po Roku 2004: Nowe Wyzwania*, pages 373–388. Wydawnictwa Naukowo-Techniczne, 2004. URL: `http://madeyski.e-informatyka.pl/download/19.pdf`. (MNiSW: 3).

Cited by:

[28.1] Tomasz Milczarek, Bartosz Sakowicz, and Andrzej Napieralski. Job recruitment system based on java enterprise edition 5 platform. In *Modern Problems of Radio Engineering, Telecommunications and Computer Science, 2008 Proceedings of International Conference on*, pages 457 –460, feb. 2008.

[29] Przemysław Mielcarek, Marcin Parczewski, and Lech Madeyski. E-learning — analiza celów i możliwości ich realizacji na podstawie istniejących specyfikacji i standardów. *Prace Naukowe Uniwersytetu Ekonomicznego (dawniej Akademii Ekonomicznej) we Wrocławiu*, 1044:548–556, Oficyna Uniwersytetu Ekonomicznego (dawniej Akademii Ekonomicznej) we Wrocławiu, 2004. URL: `http://madeyski.e-informatyka.pl/download/20.pdf`. (MNiSW: 9).

[30] Lech Madeyski and Marcin Kruczkiewicz. Rozszerzalny Obiektowy Model Projektu Informatycznego. *Prace Naukowe Uniwersytetu Ekonomicznego (dawniej Akademii Ekonomicznej) we Wrocławiu*, 1044:398–407, Oficyna Uniwersytetu Ekonomicznego (dawniej Akademii Ekonomicznej) we Wrocławiu, 2004. URL: `http://madeyski.e-informatyka.pl/download/21.pdf`. (MNiSW: 9).

[31] Lech Madeyski and Piotr Karwaczyński. Działania Projakościowe w Procesie Wytwarzania Oprogramowania. *Prace Naukowe Uniwersytetu Ekonomicznego (dawniej Akademii Ekonomicznej) we Wrocławiu*, 1044:389–397, Oficyna Uniwersytetu Ekonomicznego (dawniej Akademii Ekonomicznej) we Wrocławiu, 2004. URL: `http://madeyski.e-informatyka.pl/download/22.pdf`. (MNiSW: 9).

[32] Lech Madeyski and Marcin Kubasiak. Zwinna Specyfikacja Wymagań. In Zbigniew Huzar and Zygmunt Mazur, editors, *Problemy i metody inżynierii oprogramowania*, pages 53–68. Wydawnictwa Naukowo-Techniczne, 2003. URL: `http://madeyski.e-informatyka.pl/download/15.pdf`.

[33] Lech Madeyski. XML w bazach danych. *Bazy Danych — Prace Naukowe Wydziałowego Zakładu Informatyki Politechniki Wrocławskiej*, (4):81–89, 2003.

[34] Lech Madeyski and Paweł Mazur. Nowoczesne aplikacje internetowe. *Telenet Forum*, (5):14–17, 2003. URL: `http://madeyski.e-informatyka.pl/download/17.pdf`.

[35] Lech Madeyski. XML w bazach danych. Raport Serii PRE Z09/02/P-006, Wydziałowy Zakład Informatyki Politechniki Wrocławskiej, 2002.

[36] Lech Madeyski. Nowe koncepcje tworzenia aplikacji internetowych na przykładzie portalu E-INFORMATYKA.PL. *Prace Naukowe Uniwersytetu Ekonomicznego (dawniej Akademii Ekonomicznej) we Wrocławiu*, 955:425–437, Oficyna Uniwersytetu Ekonomicznego (dawniej Akademii Ekonomicznej) we Wrocławiu, 2002. URL: `http://madeyski.e-informatyka.pl/download/14.pdf`. (MNiSW: 9).

[37] Lech Madeyski. Koncepcja Integracji Technologii J2EE, XML, XSLT i Wzorów Projektowych w Aplikacjach Internetowych. Raport Serii PRE Z09/01/P-020, Wydziałowy Zakład Informatyki, Politechnika Wrocławska, 2001.

[38] Lech Madeyski. Portale Informacyjne na Platformie Java. In Andrzej Małachowski, editor, *Komunikacja Gospodarcza. Studia i Materiały.*, pages 113–118. Oficyna Akademii Ekonomicznej we Wrocławiu, 2001. (MNiSW: 3).

[39] Lech Madeyski. Portale Informacyjne na Platformie Java. Raport Serii PRE Z09/01/P-006, Wydziałowy Zakład Informatyki Politechniki Wrocławskiej, 2001.

[40] Lech Madeyski. Prognozy Technologiczne w Dziedzinie Internetu. Raport Serii PRE. Z09/01/P-008. Referat wygłoszony na konferencji "Prognozy technologiczne w dziedzinie energetyki, środowiska i internetu". Wrocław, WCTT 30 marca 2001., Wydziałowy Zakład Informatyki Politechniki Wrocławskiej, 2001. Referat wygłoszony na konferencji "Prognozy technologiczne w dziedzinie energetyki, środowiska i internetu". Wrocław, WCTT 30 marca 2001.

[41] Lech Madeyski. Wydajnościowo-niezawodnościowa analiza warstwy szkieletowej sieci komputerowych (prolegomena). *Pro Dialog*, (10):59–65, 2000. URL: `http://madeyski.e-informatyka.pl/download/8.pdf`. (MNiSW: 2).

[42] Lech Madeyski. Wyznaczanie pesymistycznej złożoności obliczeniowej algorytmu faktoryzacji fact. Raport Serii PRE Z09/99/P-003, Wydziałowy Zakład Informatyki, Politechnika Wrocławska, 1999.

[43] Lech Madeyski. Zmodyfikowana technika programowania dynamicznego. *Pro Dialog*, (9):33–40, 1999. URL: `http://madeyski.e-informatyka.pl/download/5.pdf`. (MNiSW: 2).

[44] Lech Madeyski. Pesymistyczna złożoność obliczeniowa algorytmu faktoryzacji *Fact*. *Pro Dialog*, (9):41–54, 1999. URL: `http://madeyski.e-informatyka.pl/download/6.pdf`. (MNiSW: 2).

[45] Lech Madeyski. *Metody obliczania niezawodności sieci K-terminali*. Rozprawa doktorska - Wydziałowy Zakład Informatyki Politechniki Wrocławskiej 1999 Seria PRE nr 20, Politechnika Wrocławska, Wydział Informatyki i Zarządzania, Wrocław, Wyb.Wyspiańskiego 27, 50-370 Wrocław, Polska, listopad 1999.

[46] Lech Madeyski and Zygmunt Mazur. Nowy algorytm do szybkiego obliczania niezawodności sieci. *Zagadnienia Eksploatacji Maszyn*, 33(3):391–404, 1998. URL: `http://madeyski.e-informatyka.pl/download/1.pdf`. (MNiSW: 6).

[47] Lech Madeyski and Zygmunt Mazur. Wykrywanie równoważnych podproblemów w algorytmie faktoryzacji do obliczania niezawodności K-terminali. In *Systemy Informatyczne Cz.1. Materiały dydaktyczno-naukowe.*, pages 73–81. Politechnika Czestochowska, 1998.

[48] Lech Madeyski and Zygmunt Mazur. Nowa technika programowania dynamicznego. Raport Serii PRE Z09/98/P-005, Wydziałowy Zakład Informatyki, Politechnika Wrocławska, 1998.

Liczba cytowań: 84
Liczba cytowań z listy filadelfijskiej: 28 (w tym jedna pozycja bez IF)