

Primary studies reviewed in the systematic literature review (SLR) on Software-testing education

<p>Vahid Garousi Wageningen University, Wageningen, Netherlands vahid.garousi@wur.nl</p>	<p>Austen Rainer Queen's University Belfast, Northern Ireland, UK a.rainer@qub.ac.uk</p>	<p>Per Lauvås Kristiania University College, Oslo, Norway per.lauvas@kristiania.no</p>	<p>Andrea Arcuri Kristiania University College, Oslo, Norway andrea.arcuri@kristiania.no</p>
------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------

[P1]	S. Tiwari, V. Saini, P. Singh, and A. Sureka, "A case study on the application of case-based learning in software testing," in Proceedings of Innovations in Software Engineering Conference, 2018, p. 11.
[P2]	W. Zheng, Y. Bai, and H. Che, "A computer-assisted instructional method based on machine learning in software testing class," Computer Applications in Engineering Education, vol. 26, no. 5, pp. 1150-1158, 2018.
[P3]	K. Buffardi and S. H. Edwards, "A formative study of influences on student testing behaviors," in Proceedings of ACM technical symposium on Computer science education, 2014, pp. 597-602.
[P4]	S. K. Sowe, I. Stamelos, and I. Deligiannis, "A framework for teaching software testing using F/OSS methodology," in IFIP International Conference on Open Source Systems, 2006, pp. 261-266.
[P5]	S. Sheth, J. Bell, and G. Kaiser, "A Gameful Approach to Teaching Software Design and Software Testing," Computer Games and Software Engineering, vol. 9, p. 91, 2015.
[P6]	M. H. Goldwasser, "A gimmick to integrate software testing throughout the curriculum," in ACM SIGCSE Bulletin, 2002, vol. 34, no. 1, pp. 271-275.
[P7]	F. B. V. Benitti, "A Methodology to Define Learning Objects Granularity: A Case Study in Software Testing," Informatics in Education, vol. 17, no. 1, pp. 1-20, 2018.
[P8]	V. Ramasamy, H. Alomari, J. Kiper, and G. Potvin, "A minimally disruptive approach of integrating testing into computer programming courses," in International Workshop on Software Engineering Education for Millennials, 2018, pp. 1-7.
[P9]	C. Desai, "A pedagogical approach to introducing test-driven development," Master of Science thesis, Faculty of California Polytechnic State University, 2008.
[P10]	E. L. Jones and C. L. Chatmon, "A perspective on teaching software testing," Journal of Computing Sciences in Colleges, vol. 16, no. 3, pp. 92-100, 2001.
[P11]	G. Fraser, A. Gambi, and J. M. Rojas, "A preliminary report on gamifying a software testing course with the Code Defenders testing game," in Proceedings of European Conference of Software Engineering Education, 2018, pp. 50-54.
[P12]	A. Gaspar, S. Langevin, N. Boyer, and R. Tindell, "A preliminary review of undergraduate programming students' perspectives on writing tests, working with others, & using peer testing," in Proceedings of the ACM SIGITE conference on Information technology education, 2013, pp. 109-114.
[P13]	N. Silvis-Cividjian, "A safety-aware, systems-based approach to teaching software testing," in Proceedings of ACM Conference on Innovation and Technology in Computer Science Education, 2018, pp. 314-319.
[P14]	Z. Zakaria, "A State-of-Practice on Teaching Software Verification and Validation," in Conf. of the American Society for Engineering Education, 2009.
[P15]	L. P. Scatolon, M. L. Fioravanti, J. M. Prates, R. E. Garcia, and E. F. Barbosa, "A survey on graduates curriculum-based knowledge gaps in software testing," in Annual Frontiers in Education Conference, 2018.
[P16]	J. L. Whalley and A. Philpott, "A unit testing approach to building novice programmers' skills and confidence," in Proceedings of the Australasian Computing Education Conference, 2011, pp. 113-118.
[P17]	A. Gaspar and S. Langevin, "Active learning in introductory programming courses through student-led "live coding" and test-driven pair programming," in International Conference on Education and Information Systems, Technologies and Applications, 2007.
[P18]	C. Brown, R. Pastel, M. Seigel, C. Wallace, and L. Ott, "Adding unit test experience to a usability centered project course," in Proceedings of the technical symposium on Computer science education, 2014, pp. 259-264.
[P19]	T. Hynninen, A. Knutas, and J. Kasurinen, "Aligning Software Testing Activities to V-Model Phases," in Proceedings of the Koli Calling International Conference on Computing Education Research, 2018, p. 28.
[P20]	M. Allison and S. F. Joo, "An adaptive delivery strategy for teaching software testing and maintenance," in International Conference on Computer Science & Education, 2015, pp. 237-242.
[P21]	B. Carlson, "An agile classroom experience: Teaching TDD and refactoring," in Agile 2008 Conference, 2008, pp. 465-469.
[P22]	O. Ochoa and S. Salamah, "An approach to enhance students' competency in software verification techniques," in IEEE Frontiers in Education Conference, 2015, pp. 1-9.
[P23]	R. Smith, T. Tang, J. Warren, and S. Rixner, "An Automated system for interactively learning software testing," in Proceedings of ACM Conference on Innovation and Technology in Computer Science Education, 2017, pp. 98-103.

[P24]	J. Collofello and K. Vehathiri, "An environment for training computer science students on software testing," in Proceedings Frontiers in Education Conference, 2005, pp. T3E-6.
[P25]	D. S. Janzen, J. Clements, and M. Hilton, "An evaluation of interactive test-driven labs with WebIDE in CS0," in Proceedings of the International Conference on Software Engineering, 2013, pp. 1090-1098.
[P26]	E. G. Barriocanal, M.-Á. S. Urbán, I. A. Cuevas, and P. D. Parez, "An experience in integrating automated unit testing practices in an introductory programming course," ACM SIGCSE Bulletin, vol. 34, no. 4, pp. 125-128, 2002.
[P27]	E. F. Barbosa, S. d. R. S. de Souza, and J. C. Maldonado, "An experience on applying learning mechanisms for teaching inspection and software testing," in Conference on Software Engineering Education and Training, 2008, pp. 189-196.
[P28]	M. A. Brito, J. L. Rossi, S. R. de Souza, and R. T. Braga, "An experience on applying software testing for teaching introductory programming courses," CLEI Electronic Journal, vol. 15, no. 1, pp. 5-5, 2012.
[P29]	E. L. Jones, "An experiential approach to incorporating software testing into the computer science curriculum," in Conference Proceedings of Frontiers in Education Conference. Impact on Engineering and Science Education, 2001, vol. 2, pp. F3D-7.
[P30]	A. Soska, J. Mottok, and C. Wolff, "An experimental card game for software testing: Development, design and evaluation of a physical card game to deepen the knowledge of students in academic software testing education," in IEEE Global Engineering Education Conference, 2016, pp. 576-584.
[P31]	J. R. Barbosa, P. Valle, J. Maldonado, M. Delamaro, and A. M. Vincenzi, "An experimental evaluation of peer testing in the context of the teaching of software testing," in International Symposium on Computers in Education, 2017, pp. 1-6.
[P32]	V. Thurner and A. Böttcher, "An "objects first, tests second" approach for software engineering education," in IEEE Frontiers in Education Conference, 2015, pp. 1-5.
[P33]	V. Garousi, "An open modern software testing laboratory courseware: an experience report," in IEEE Conference on Software Engineering Education and Training, 2010, pp. 177-184.
[P34]	P. Lauves, "Analyzing student code after introducing portfolio assessment and automated testing," in Norwegian Conference on Organizations' Use of IT (NOKOBIT), 2015, vol. 23, no. 1.
[P35]	S. M. Rahman, "Applying the tbc method in introductory programming courses," in Annual Frontiers In Education Conference-Global Engineering, 2007, pp. T1E-20-T1E-21.
[P36]	A. Čaušević, "Appreciate the journey not the destination-Using video assignments in software testing education," in Proceedings of the Workshops of the German Software Engineering Conference, 2018, vol. 2066, pp. 4-7.
[P37]	J. R. Horgan and A. P. Mathur, "Assessing testing tools in research and education," IEEE Software, vol. 9, no. 3, pp. 61-69, 1992.
[P38]	J. Bowyer and J. Hughes, "Assessing undergraduate experience of continuous integration and test-driven development," in Proceedings of the international conference on Software engineering, 2006, pp. 691-694.
[P39]	C. Kaner, "Assessment in the software testing course," in Workshop on the Teaching of Software Testing, 2003.
[P40]	Z. Shams, "Automated assessment of students' testing skills for improving correctness of their code," in Proceedings of companion publication for conference on Systems, programming, & applications: software for humanity, 2013, pp. 37-40.
[P41]	R. Bryce et al., "Bug catcher: a system for software testing competitions," in Proceeding of the ACM technical symposium on Computer science education, 2013, pp. 513-518.
[P42]	K. Buffardi and P. Valdivia, "Bug Hide-and-Seek: An Educational Game for Investigating Verification Accuracy in Software Tests," in IEEE Frontiers in Education.
[P43]	S. Elbaum, S. Person, J. Dokulil, and M. Jorde, "Bug hunt: Making early software testing lessons engaging and affordable," in Proceedings of the international conference on Software Engineering, 2007, pp. 688-697.
[P44]	D. E. Krutz and M. Lutz, "Bug of the day: Reinforcing the importance of testing," in IEEE Frontiers in Education Conference, 2013, pp. 1795-1799.
[P45]	G. Joshi and P. Desai, "Building software testing skills in undergraduate students using spiral model approach," in International conference on technology for education, 2016, pp. 244-245.
[P46]	S. Hooshangi, R. Weiss, and J. Cappos, "Can the security mindset make students better testers?," in Proceedings of the ACM Technical Symposium on Computer Science Education, 2015, pp. 404-409.
[P47]	Z. Shams and S. H. Edwards, "Checked coverage and object branch coverage: New alternatives for assessing student-written tests," in Proceedings of the ACM Technical Symposium on Computer Science Education, 2015, pp. 534-539.
[P48]	J. M. Rojas and G. Fraser, "Code defenders: a mutation testing game," in IEEE International Conference on Software Testing, Verification and Validation Workshops, 2016, pp. 162-167.
[P49]	P. J. Clarke, J. Pava, Y. Wu, and T. M. King, "Collaborative web-based learning of testing tools in SE courses," in Proceedings of ACM technical symposium on Computer science education, 2011, pp. 147-152.
[P50]	Z. Chen, A. Memon, and B. Luo, "Combining research and education of software testing: a preliminary study," in Proceedings of the ACM Symposium on Applied Computing, 2014, pp. 1179-1180.
[P51]	V. Lappalainen, J. Itkonen, V. Isomettinen, and S. Kollanus, "ComTest: a tool to impart TDD and unit testing to introductory level programming," in Proceedings of the conference on Innovation and technology in computer science education, 2010, pp. 63-67.

[P52]	P. H. D. Valle, E. F. Barbosa, and J. C. Maldonado, "CS curricula of the most relevant universities in Brazil and abroad: Perspective of software testing education," in International Symposium on Computers in Education, 2015, pp. 62-68.
[P53]	V. Isomäki and J. Lappalainen, "CS1 with games and an emphasis on TDD and unit testing: piling a trend upon a trend," ACM Inroads, vol. 3, no. 3, pp. 62-68, 2012.
[P54]	V. Garousi and A. Mathur, "Current state of the software testing education in north American academia and some recommendations for the new educators," in IEEE Conference on Software Engineering Education and Training, 2010, pp. 89-96.
[P55]	B. Zhu and S. Zhang, "Curriculum reform and practice of software testing," in International Conference on Education Technology and Information System, 2013.
[P56]	R. Mao, M. Wang, X. Wang, and Y. Zhang, "Curriculum Reform on Software Testing Courses to Strengthen Abilities of Engineering Practice," International Journal of Science, vol. 4, no. 10, pp. 16-19, 2017.
[P57]	G. Lopez, F. Cocozza, A. Martinez, and M. Jenkins, "Design and implementation of a software testing training course," in ASEE Annual Conference & Exposition, 2015.
[P58]	M. J. A. M. J. Rodriguez and E. Rojas, "Designing a blended software testing course for embedded C software engineers," in International Conference on Computers and Advanced Technology in Education.
[P59]	R. Agarwal, S. H. Edwards, and M. A. Pérez-Quñones, "Designing an adaptive learning module to teach software testing," in ACM SIGCSE Bulletin, 2006, vol. 38, no. 1, pp. 259-263.
[P60]	D. Middleton, "Developing students' testing skills: covering space: nifty assignment," Journal of Computing Sciences in Colleges, vol. 30, no. 5, pp. 29-31, 2015.
[P61]	D. Middleton, "Developing students' testing skills: distinguishing functions," Journal of Computing Sciences in Colleges, vol. 28, no. 5, pp. 73-74, 2013.
[P62]	A. Alelaiwi, "Direct assessment methodology for a software testing course," Life Science Journal, vol. 11, no. 6s, 2014.
[P63]	S. H. Edwards and Z. Shams, "Do student programmers all tend to write the same software tests?," in Proceedings of conference on Innovation & technology in computer science education, 2014, pp. 171-176.
[P64]	P. H. D. Valle, A. M. Toda, E. F. Barbosa, and J. C. Maldonado, "Educational games: A contribution to software testing education," in IEEE Frontiers in Education Conference, 2017, pp. 1-8.
[P65]	K. Buffardi and S. H. Edwards, "Effective and ineffective software testing behaviors by novice programmers," in Proceedings of the ACM conference on International computing education research, 2013, pp. 83-90.
[P66]	M. Thiry, A. Zoucas, and A. C. da Silva, "Empirical study upon software testing learning with support from educational game," in Proceedings of the International Conference on Software Engineering & Knowledge Engineering, 2011, pp. 481-484.
[P67]	R. Pham, S. Kiesling, O. Liskin, L. Singer, and K. Schneider, "Enablers, inhibitors, and perceptions of testing in novice software teams," in Proceedings of the ACM SIGSOFT International Symposium on Foundations of Software Engineering, 2014, pp. 30-40.
[P68]	J. J. Li and P. Morreale, "Enhancing CS1 curriculum with testing concepts: a case study," Journal of Computing Sciences in Colleges, vol. 31, no. 3, pp. 36-43, 2016.
[P69]	J. Maldonado and E. Barbosa, "Establishing a mutation testing educational module based on IMA-CID," in Workshop on Mutation Analysis, part of ISSRE Conf., 2006, pp. 14-14.
[P70]	J. C. Carver and N. A. Kraft, "Evaluating the testing ability of senior-level computer science students," in IEEE-CS Conference on Software Engineering Education and Training, 2011, pp. 169-178.
[P71]	R. A. Oliveira, L. B. Oliveira, B. B. Cafeo, and V. H. Durelli, "Evaluation and assessment of effects on exploring mutation testing in programming courses," in IEEE Frontiers in Education Conference, 2015, pp. 1-9.
[P72]	O. A. L. Lemos, F. C. Ferrari, F. F. Silveira, and A. Garcia, "Experience report: Can software testing education lead to more reliable code?," in IEEE International Symposium on Software Reliability Engineering, 2015, pp. 359-369.
[P73]	T. Y. Chen and P.-L. Poon, "Experience with teaching black-box testing in a computer science/software engineering curriculum," IEEE Transactions on Education, vol. 47, no. 1, pp. 42-50, 2004.
[P74]	J. Kasurinen, "Experiences from a web-based course in software testing and quality assurance," International Journal of Computer Applications, vol. 166, no. 2, 2017.
[P75]	J. Timoney, S. Brown, and D. Ye, "Experiences in software testing education: some observations from an international cooperation," in International Conference for Young Computer Scientists, 2008, pp. 2686-2691.
[P76]	S. H. Edwards and M. A. Pérez-Quñones, "Experiences using test-driven development with an automated grader," Journal of Computing Sciences in Colleges, vol. 22, no. 3, pp. 44-50, 2007.
[P77]	J. Spacco, D. Hovemeyer, W. Pugh, F. Emad, J. K. Hollingsworth, and N. Padua-Perez, "Experiences with marmoset: designing and using an advanced submission and testing system for programming courses," in ACM Bulletin, 2006, vol. 38, no. 3, pp. 13-17.
[P78]	Z. Bin and Z. Shiming, "Experiment teaching reform for software testing course based on CDIO," in International Conference on Computer Science & Education, 2014, pp. 488-491.
[P79]	D. Mishra, S. Ostrovska, and T. Hacaloglu, "Exploring and expanding students' success in software testing," Information Technology & People, vol. 30, no. 4, pp. 927-945, 2017.
[P80]	K. Buffardi and S. H. Edwards, "Exploring influences on student adherence to test-driven development," in Proceedings of ACM conference on Innovation and technology in computer science education, 2012, pp. 105-110.

[P81]	B. Tomia and S. Vlajia, "Functional testing for students: a practical approach," ACM SIGCSE Bulletin, vol. 40, no. 4, pp. 58-62, 2008.
[P82]	Y. Fu and P. Clarke, "Gamification based cyber enabled learning environment of software testing," in Proceedings of ASEE Annual Conference and Expo, 2016.
[P83]	G. Fraser, A. Gambi, M. Kreis, and J. M. Rojas, "Gamifying a Software Testing Course with Code Defenders," in ACM Technical Symposium on Computer Science Education, 2019.
[P84]	T. Hynninen, J. Kasurinen, A. Knutas, and O. Taipale, "Guidelines for software testing education objectives from industry practices with a constructive alignment approach," in Proceedings of ACM Conference on Innovation and Technology in Computer Science Education, 2018, pp. 278-283.
[P85]	J. Spacco and W. Pugh, "Helping students appreciate test-driven development (TDD)," in ACM SIGPLAN symposium on Object-oriented programming systems, languages, and applications, 2006, pp. 907-913.
[P86]	P. Zhang, J. White, and D. C. Schmidt, "HoliCoW: automatically breaking team-based software projects to motivate student testing," in IEEE/ACM International Conference on Software Engineering Companion, 2016, pp. 436-439.
[P87]	A. Allowatt and S. H. Edwards, "IDE Support for test-driven development and automated grading in both Java and C++," in Proceedings of the OOPSLA workshop on Eclipse technology eXchange, 2005, pp. 100-104.
[P88]	O. S. Gómez, S. Vegas, and N. Juristo, "Impact of CS programs on the quality of test cases generation: An empirical study," in Proceedings of the International Conference on Software Engineering Companion, 2016, pp. 374-383.
[P89]	P. J. Clarke, D. L. Davis, R. Chang-Lau, and T. M. King, "Impact of using tools in an undergraduate software testing course supported by WRESTT," ACM Transactions on Computing Education (TOCE), vol. 17, no. 4, p. 18, 2017.
[P90]	K. Buffardi and S. H. Edwards, "Impacts of adaptive feedback on teaching test-driven development," in Proceeding of the ACM technical symposium on Computer science education, 2013, pp. 293-298.
[P91]	K. Buffardi and S. H. Edwards, "Impacts of Teaching test-driven development to Novice Programmers," International Journal of Information and Computer Science, vol. 1, no. 6, p. 9, 2012.
[P92]	Z. Yinnan and W. Xiaochi, "Implementation of Software Testing Course Based on CDIO," in International Conference on Computer Science & Education, 2011, pp. 107-110.
[P93]	C. Desai, D. S. Janzen, and J. Clements, "Implications of integrating test-driven development into CS1/CS2 curricula," in ACM SIGCSE Bulletin, 2009, vol. 41, no. 1, pp. 148-152.
[P94]	R. Aziz, "Improving high order thinking skills in software testing course," International Journal of Computer Science and Information Security, vol. 14, no. 8, p. 966, 2016.
[P95]	I. Alazzam and M. Akour, "Improving software testing course experience with pair testing pattern," International Journal of Teaching and Case Studies, vol. 6, no. 3, pp. 244-250, 2015.
[P96]	S. H. Edwards, "Improving student performance by evaluating how well students test their own programs," Journal on Educational Resources in Computing (JERIC), vol. 3, no. 3, p. 1, 2003.
[P97]	V. Subbian, N. Niu, and C. C. Purdy, "Inclusive and evidence-based instruction in software testing education," in Annual conference of the American Society for Engineering Education, 2016.
[P98]	V. Garousi, "Incorporating real-world industrial testing projects in software testing courses: Opportunities, challenges, and lessons learned," in IEEE-CS Conference on Software Engineering Education and Training, 2011, pp. 396-400.
[P99]	S. K. Bajaj and S. Balram, "Incorporating software testing as a discipline in curriculum of computing courses," in International United Information Systems Conference, 2009, pp. 404-410.
[P100]	E. F. Barbosa, M. A. Silva, C. K. Corte, and J. C. Maldonado, "Integrated teaching of programming foundations and software testing," in Annual Frontiers in Education Conference, 2008, pp. S1H-5-S1H-10.
[P101]	S. Frezza, "Integrating testing and design methods for undergraduates: teaching software testing in the context of software design," in Annual Frontiers in Education, 2002, vol. 3, pp. S1G-S1G.
[P102]	P. Yujian Fu, P. J. Clarke, and N. Barnes Jr, "Integrating Software Testing to CS Curriculum Using WRESTT-CyLE," in Annual Conference of the American Society for Engineering Education, 2015.
[P103]	P. J. Clarke, D. Davis, T. M. King, J. Pava, and E. L. Jones, "Integrating testing into software engineering courses supported by a collaborative learning environment," ACM Transactions on Computing Education, vol. 14, no. 3, p. 18, 2014.
[P104]	E. L. Jones, "Integrating testing into the curriculum-arsenic in small doses," ACM SIGCSE Bulletin, vol. 33, no. 1, pp. 337-341, 2001.
[P105]	E. F. Barbosa, J. C. Maldonado, R. LeBlanc, and M. Guzdial, "Introducing testing practices into objects and design course," in Proceedings Conference on Software Engineering Education and Training, 2003, pp. 279-286.
[P106]	A. Patterson, M. Källing, and J. Rosenberg, "Introducing unit testing with BlueJ," ACM SIGCSE Bulletin, vol. 35, no. 3, pp. 11-15, 2003.
[P107]	V. K. Proulx and R. Rasala, "Java IO and testing made simple," in ACM SIGCSE Bulletin, 2004, vol. 36, no. 1, pp. 161-165.
[P108]	H. Tråtteberg and T. Aalberg, "JExercise: a specification-based and test-driven exercise support plugin for Eclipse," in OOPSLA workshop on eclipse technology eXchange, 2006, vol. 22, no. 23, pp. 70-74.
[P109]	M. Wahid and A. Almalaise, "JUnit framework: An interactive approach for basic unit testing learning in Software Engineering," in International Congress on Engineering Education, 2011, pp. 159-164.
[P110]	B. Talon, D. Lecllet, A. Lewandowski, and G. Bourguin, "Learning software testing using a collaborative activities oriented platform," in IEEE International Conference on Advanced Learning Technologies, 2009, pp. 443-445.

[P111]	J. Snyder, S. H. Edwards, and M. A. Pérez-Quiñones, "LIFT: taking GUI unit testing to new heights," in Proceedings of ACM technical symposium on Computer science education, 2011, pp. 643-648.
[P112]	Q. Li and B. W. Boehm, "Making winners for both education and research: Verification and validation process improvement practice in a software engineering course," in IEEE-CS Conference on Software Engineering Education and Training, 2011, pp. 304-313.
[P113]	D. Towey, T. Y. Chen, F.-C. Kuo, H. Liu, and Z. Q. Zhou, "Metamorphic testing: A new student engagement approach for a new software testing paradigm," in IEEE International Conference on Teaching, Assessment, and Learning for Engineering, 2016, pp. 218-225.
[P114]	T. Shepard, M. Lamb, and D. Kelly, "More testing should be taught," Communications of the ACM, vol. 44, no. 6, pp. 103-108, 2001.
[P115]	K. Aaltonen, P. Ihantola, and O. Seppälä, "Mutation analysis vs. code coverage in automated assessment of students' testing skills," in Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion, 2010, pp. 153-160.
[P116]	M. Hilton and D. S. Janzen, "On teaching arrays with test-driven learning in WebIDE," in Proceedings of ACM annual conference on Innovation and technology in computer science education, 2012, pp. 93-98.
[P117]	T. Shepard, "On teaching software verification and validation," in Conference on Software Engineering Education, 1995, pp. 375-385.
[P118]	R. A. Oliveira, L. Oliveira, B. B. Cafeo, and V. Durelli, "On using mutation testing for teaching programming to novice programmers," in International Conference on Computers in Education, 2014, pp. 394-396.
[P119]	J. Clements and D. Janzen, "Overcoming obstacles to test-driven learning on day one," in International Conference on Software Testing, Verification, and Validation Workshops, 2010, pp. 448-453.
[P120]	N. Clark, "Peer testing in software engineering projects," in Proceedings of the Australasian Conference on Computing Education, 2004, pp. 41-48.
[P121]	C. Li, "Penetration testing curriculum development in practice," Journal of Information Technology Education: Innovations in Practice, vol. 14, no. 1, pp. 85-99, 2015.
[P122]	S. A. Brian, R. N. Thomas, J. M. Hogan, and C. Fidge, "Planting Bugs: A System for Testing Students' Unit Tests," in Proceedings of the ACM Conference on Innovation and Technology in Computer Science Education, 2015, pp. 45-50.
[P123]	V. L. Neto, R. Coelho, L. Leite, D. S. Guerrero, and A. P. Mendonça, "POPT: a problem-oriented programming and testing approach for novice students," in international conference on software engineering, 2013, pp. 1099-1108.
[P124]	C. Kaner and S. Padmanabhan, "Practice and transfer of learning in the teaching of software testing," in IEEE Conference on Software Engineering Education & Training, 2007, pp. 157-166.
[P125]	M. Aniche, F. Hermans, and A. van Deursen, "Pragmatic Software Testing Education," in ACM Technical Symposium on Computer Science Education, 2019, pp. 414-420
[P126]	C. Matthies, A. Treffer, and M. Uflacker, "Prof. CI: Employing continuous integration services and Github workflows to teach test-driven development," in IEEE Frontiers in Education Conference, 2017, pp. 1-8.
[P127]	D. M. De Souza, J. C. Maldonado, and E. F. Barbosa, "ProgTest: An environment for the submission and evaluation of programming assignments based on testing activities," in IEEE-CS Conference on Software Engineering Education and Training, 2011, pp. 1-10.
[P128]	D. Almog, H. Chasidim, and S. Mark, "Quality and testing-new teaching approaches for software engineers," in World Transactions on Engineering and Technology Education, 2018, vol. 16, pp. 140-145.
[P129]	L. Fu, "Research on case teaching of software testing course with open source software," Advances in Information Sciences and Service Sciences, vol. 4, no. 13, 2012.
[P130]	S. H. Edwards, "Rethinking computer science education from a test-first perspective," in ACM conference on Object-oriented programming, systems, languages, and applications, 2003, pp. 148-155.
[P131]	A. Allevato and S. H. Edwards, "RoboLIFT: engaging CS2 students with testable, automatically evaluated android applications," in Proceedings of the ACM technical symposium on Computer Science Education, 2012, pp. 547-552.
[P132]	S. H. Edwards, Z. Shams, M. Cogswell, and R. C. Senkbeil, "Running students' software tests against each others' code: new life for an old gimmick," in Proceedings of ACM technical symposium on Computer Science Education, 2012, pp. 221-226.
[P133]	J. Bell, S. Sheth, and G. Kaiser, "Secret ninja testing with HALO software engineering," in Proceedings of the international workshop on Social software engineering, 2011, pp. 43-47.
[P134]	A. J. A. Wang, "Security testing in software engineering courses," in Annual Frontiers in Education Conf., 2004, pp. F1C-13.
[P135]	S. Šošić, O. Ristić, K. Mitrović, and D. Milošević, "Software Testing Course in IT Undergraduate Education in Serbia," Information Technology, vol. 4, no. 6, p. 8.
[P136]	F. Chan, T. Tse, W. Tang, and T. Chen, "Software testing education and training in Hong Kong," in International Conference on Quality Software, 2005, pp. 313-316.
[P137]	J. P. U. Pech, R. A. A. Vera, and O. S. GÁmez, "Software testing education through a collaborative virtual approach," in International Conference on Software Process Improvement, 2017, pp. 231-240.
[P138]	E. L. Jones, "Software testing in the computer science curriculum—a holistic approach," in Proceedings of the Australasian conference on Computing education, 2000, pp. 153-157.

[P139]	B. Hang, "Software Testing Training in Vocational Technical Education," in <i>Education and Educational Technology</i> : Springer, 2011, pp. 567-574.
[P140]	G. Fisher and C. Johnson, "Specification-Based Testing in Software Engineering Courses," in <i>Proceedings of the ACM Technical Symposium on Computer Science Education</i> , 2018, pp. 800-805.
[P141]	E. L. Jones, "SPRAE: A Framework for Teaching Software Testing in the Undergraduate Curriculum," in <i>International Workshop on Agents and Data Mining Interaction (ADMI)</i> , 2000.
[P142]	T. Cowling, "Stages in teaching software testing," in <i>Proceedings of the International Conference on Software Engineering</i> , 2012, pp. 1185-1194.
[P143]	D. L. Davis and R. C. Lau, "Student Learning and Use of Tools in an Undergraduate Software Testing Class," in <i>Annual conference of the American Society for Engineering Education</i> , vol. 24, p. 1.
[P144]	N. J. Wahl, "Student-run usability testing," in <i>Conference on Software Engineering Education and Training</i> , 2000, pp. 123-131.
[P145]	T. Dvornik, D. S. Janzen, J. Clements, and O. Dekhtyar, "Supporting introductory test-driven labs with WebIDE," in <i>IEEE-CS Conference on Software Engineering Education and Training</i> , 2011, pp. 51-60.
[P146]	M. Thornton, S. H. Edwards, R. P. Tan, and M. A. Pérez-Quiñones, "Supporting student-written tests of GUI programs," in <i>ACM SIGCSE Bulletin</i> , 2008, vol. 40, no. 1, pp. 537-541.
[P147]	H. B. Christensen, "Systematic testing should not be a topic in the computer science curriculum!," in <i>ACM SIGCSE Bulletin</i> , 2003, vol. 35, no. 3, pp. 7-10.
[P148]	R. M. Snyder, "Teacher specification and student implementation of a unit testing methodology in an introductory programming course," <i>Journal of Computing Sciences in Colleges</i> , vol. 19, no. 3, pp. 22-32, 2004.
[P149]	H. Liu, F.-C. Kuo, and T. Y. Chen, "Teaching an end-user testing methodology," in <i>IEEE Conference on Software Engineering Education and Training</i> , 2010, pp. 81-88.
[P150]	D. Hoffman, P. Strooper, and P. Walsh, "Teaching and testing," in <i>Proceedings of Conference on Software Engineering Education</i> , 1996, pp. 248-258.
[P151]	T. Xie, J. de Halleux, N. Tillmann, and W. Schulte, "Teaching and training developer-testing techniques and tool support," in <i>Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion</i> , 2010, pp. 175-182.
[P152]	T. Y. Chen, F.-C. Kuo, and Z. Q. Zhou, "Teaching automated test case generation," in <i>International Conference on Quality Software</i> , 2005, pp. 327-332.
[P153]	T. Chen and P. Poon, "Teaching black box testing," in <i>International Conference Software Engineering: Education and Practice</i> , 1998, pp. 324-329.
[P154]	S. S. Bhattacharyya, W. Plishker, A. Gupta, and C.-C. Shen, "Teaching cross-platform design and testing methods for embedded systems using DICE," in <i>Proceedings of the Workshop on Embedded Systems Education</i> , 2011, pp. 38-45.
[P155]	S. Schaub, "Teaching cs1 with web applications and test-driven development," <i>ACM SIGCSE Bulletin</i> , vol. 41, no. 2, pp. 113-117, 2009.
[P156]	C. Kaner, "Teaching domain testing: A status report," in <i>Conference on Software Engineering Education and Training</i> , 2004, pp. 112-117.
[P157]	J. M. Rojas and G. Fraser, "Teaching Mutation Testing using Gamification," in <i>European Conference on Software Engineering Education</i> , 2016.
[P158]	D. M. De Souza, S. Isotani, and E. F. Barbosa, "Teaching novice programmers using ProgTest," <i>International Journal of Knowledge and Learning</i> , vol. 10, no. 1, pp. 60-77, 2015.
[P159]	J. P. SauvÃ© and O. L. Abath Neto, "Teaching software development with ATDD and EasyAccept," in <i>ACM SIGCSE Bulletin</i> , 2008, vol. 40, no. 1, pp. 542-546.
[P160]	O. Gotel, C. Scharff, and A. Wildenberg, "Teaching software quality assurance by encouraging student contributions to an open source web-based system for the assessment of programming assignments," <i>ACM SIGCSE Bulletin</i> , vol. 40, no. 3, pp. 214-218, 2008.
[P161]	D. Carrington, "Teaching software testing," in <i>Proceedings of the Australasian conference on Computer science education</i> , 1997, pp. 59-64.
[P162]	S. Jia and C. Yang, "Teaching software testing based on CDIO," <i>World Transactions on Engineering and Technology Education</i> , vol. 11, no. 4, pp. 476-479, 2013.
[P163]	B. S. Clegg, J. M. Rojas, and G. Fraser, "Teaching software testing concepts using a mutation testing game," in <i>IEEE/ACM International Conference on Software Engineering: Software Engineering Education and Training Track</i> , 2017, pp. 33-36.
[P164]	N. B. Harrison, "Teaching software testing from two viewpoints," <i>Journal of Computing Sciences in Colleges</i> , vol. 26, no. 2, pp. 55-62, 2010.
[P165]	I. A. Buckley and W. S. Buckley, "Teaching software testing using data structures," <i>International Journal Of Advanced Computer Science And Applications</i> , vol. 8, no. 4, pp. 1-4, 2017.
[P166]	D. Mishra, T. Hacaloglu, and A. Mishra, "Teaching Software Verification and Validation Course: A Case Study," <i>International Journal of Engineering Education</i> , vol. 30, pp. 1476-1485, 2014.
[P167]	Y. Lee, D. B. Marepalli, and J. Yang, "Teaching test-drive development using dojo," <i>Journal of Computing Sciences in Colleges</i> , vol. 32, no. 4, pp. 106-112, 2017.

[P168]	M. Missiroli, D. Russo, and P. Ciancarini, "Teaching test-first programming: Assessment and solutions," in IEEE Annual Computer Software and Applications Conference, 2017, vol. 1, pp. 420-425.
[P169]	J. Paul, "Test-driven approach in programming pedagogy," Journal of Computing Sciences in Colleges, vol. 32, no. 2, pp. 53-60, 2016.
[P170]	J. Adams, "Test-driven data structures: revitalizing CS2," in ACM SIGCSE Bulletin, 2009, vol. 41, no. 1, pp. 143-147.
[P171]	V. K. Proulx, "Test-driven design for introductory OO programming," in ACM SIGCSE Bulletin, 2009, vol. 41, no. 1, pp. 138-142.
[P172]	C. G. Jones, "Test-driven development goes to school," Journal of Computing Sciences in Colleges, vol. 20, no. 1, pp. 220-231, 2004.
[P173]	D. Janzen and H. Saiedian, "Test-driven learning in early programming courses," in ACM SIGCSE Bulletin, 2008, vol. 40, no. 1, pp. 532-536.
[P174]	D. S. Janzen and H. Saiedian, "Test-driven learning: intrinsic integration of testing into the CS/SE curriculum," in ACM SIGCSE Bulletin, 2006, vol. 38, no. 1, pp. 254-258.
[P175]	M. Ricken and R. Cartwright, "Test-first Java concurrency for the classroom," in Proceedings of the ACM technical symposium on Computer science education, 2010, pp. 219-223.
[P176]	C. Leska, "Testing across the curriculum: square one!," Journal of Computing Sciences in Colleges, vol. 19, no. 5, pp. 163-169, 2004.
[P177]	W. Marrero and A. Settle, "Testing first: emphasizing testing in early programming courses," in ACM SIGCSE Bulletin, 2005, vol. 37, no. 3, pp. 4-8.
[P178]	J. A. Rosiene and C. Pe Rosiene, "Testing in the'small'," Journal of Computing Sciences in Colleges, vol. 19, no. 2, pp. 314-318, 2003.
[P179]	E. Scott, A. Zadirov, S. Feinberg, and R. Jayakody, "The alignment of software testing skills of IS students with industry practices: South African perspective," Journal of Information Technology Education: Research, vol. 3, pp. 161-172, 2004.
[P180]	D. H. Steinberg, "The effect of unit tests on entry points, coupling and cohesion in an introductory java programming course," in XP Universe, 2001, vol. 8.
[P181]	T. Astigarraga, E. M. Dow, C. Lara, R. Prewitt, and M. R. Ward, "The emerging role of software testing in curricula," in IEEE Conf. on Transforming Engineering Education 2010, pp. 1-26.
[P182]	O. A. L. Lemos, F. F. Silveira, F. C. Ferrari, and A. Garcia, "The impact of Software Testing education on code reliability: An empirical assessment," Journal of Systems and Software, vol. 137, pp. 497-511, 2018.
[P183]	L. Palmer, "The inclusion of a software testing module in the Information Systems Honours course," South African Computer Journal, vol. 12, no. 1, pp. 83-86, 2008.
[P184]	C. D. Allison, "The simplest unit test tool that could possibly work," Journal of Computing Sciences in Colleges, vol. 23, no. 1, pp. 183-189, 2007.
[P185]	D. I. Alkadi and G. Alkadi, "To C++ or to Java, that is the question! Featuring a test plan and an automated testing assistant for object oriented testing," in Proceedings of Aerospace Conference 2002, vol. 7, pp. 7-3679.
[P186]	G. Braught and J. Midkiff, "Tool design and student testing behavior in an introductory java course," in Proceedings of the ACM Technical Symposium on Computing Science Education, 2016, pp. 449-454.
[P187]	T. Briggs and C. D. Girard, "Tools and techniques for test-driven learning in CS1," Journal of Computing Sciences in Colleges, vol. 22, no. 3, pp. 37-43, 2007.
[P188]	Z. Shams and S. H. Edwards, "Toward practical mutation analysis for evaluating the quality of student-written software tests," in Proceedings of the international ACM conference on International computing education research, 2013, pp. 53-58.
[P189]	C. Mao, "Towards a question-driven teaching method for software testing course," in International Conference on Computer Science and Software Engineering, 2008, vol. 5, pp. 645-648.
[P190]	L. P. Scatolon, J. M. Prates, D. M. De Souza, E. F. Barbosa, and R. E. Garcia, "Towards the role of test design in programming assignments," in IEEE Conference on Software Engineering Education and Training, 2017, pp. 170-179.
[P191]	D. M. de Souza, B. H. Oliveira, J. C. Maldonado, S. R. Souza, and E. F. Barbosa, "Towards the use of an automatic assessment system in the teaching of software testing," in IEEE Frontiers in Education Conference, 2014, pp. 1-8.
[P192]	D. Blaheta, "Unci: a C++-based unit-testing framework for intro students," in Proceedings of the ACM Technical Symposium on Computer Science Education, 2015, pp. 475-480.
[P193]	A. Martinez, "Use of JiTT in a graduate software testing course: an experience report," in Proceedings of the International Conference on Software Engineering: Software Engineering Education and Training, 2018, pp. 108-115.
[P194]	G. Lopez and A. Martinez, "Use of Microsoft Testing Tools to Teach Software Testing: An Experience Report," in Proceedings of the American Society for Engineering Education Annual Conference and Exposition, 2014.
[P195]	D. E. Krutz, S. A. Malachowsky, and T. Reichlmayr, "Using a real world project in a software testing course," in Proceedings of the 45th ACM technical symposium on Computer science education, 2014, pp. 49-54.
[P196]	P. J. Clarke, A. A. Allen, T. M. King, E. L. Jones, and P. Natesan, "Using a web-based repository to integrate testing tools into programming courses," in Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion, 2010, pp. 193-200.

[P197]	J. Offutt, N. Li, P. Ammann, and W. Xu, "Using abstraction and Web applications to teach criteria-based test design," in IEEE-CS Conference on Software Engineering Education and Training, 2011, pp. 227-236.
[P198]	S. Edwards, "Using Industrial Tools to Test and Grade Resolve/C++ Programs," Blacksburg, VA March 22-23, 2006, p. 6, 2006.
[P199]	J. Smith, J. Tessler, E. Kramer, and C. Lin, "Using peer review to teach software testing," in Proceedings of the international conference on International computing education research, 2012, pp. 93-98.
[P200]	S. H. Edwards, "Using software testing to move students from trial-and-error to reflection-in-action," in ACM SIGCSE Bulletin, 2004, vol. 36, no. 1, pp. 26-30.
[P201]	S. H. Edwards, "Using test-driven development in the classroom: Providing students with automatic, concrete feedback on performance," in Proceedings of the international conference on education and information systems: technologies and applications, 2003
[P202]	M. Wick, D. Stevenson, and P. Wagner, "Using testing and JUnit across the curriculum," ACM SIGCSE Bulletin, vol. 37, no. 1, pp. 236-240, 2005.
[P203]	P. J. Clarke, J. Pava, D. Davis, F. Hernandez, and T. M. King, "Using WReSTT in SE courses: An empirical study," in Proceedings of the ACM technical symposium on Computer Science Education, 2012, pp. 307-312.
[P204]	A. R. Shah, "Web-cat: A web-based center for automated testing," Masters Theses, Virginia Tech, 2003.