

REFERENSI

- [1] B. Priyambadha and F. Pradana, “Penilaian Otomatis Pada Media Pembelajaran Pemrograman Online Dengan Pendekatan Clone Code Detection,” Jul. 2018.
- [2] K. Danutama and I. Liem, “Scalable Autograder and LMS Integration,” *Procedia Technology*, vol. 11, pp. 388–395, 2013, doi: 10.1016/j.protcy.2013.12.207.
- [3] R. Harimurti, A. Iwan Nurhidayat, and Asmunin, “Building Automatic Grading Tools for Basic of Programming Lab in an Academic Institution,” in *IOP Conference Series: Materials Science and Engineering*, Apr. 2018, vol. 336, no. 1. doi: 10.1088/1757-899X/336/1/012037.
- [4] S. Nidhra, “Black Box and White Box Testing Techniques - A Literature Review,” *International Journal of Embedded Systems and Applications*, vol. 2, no. 2, pp. 29–50, Jun. 2012, doi: 10.5121/ijesa.2012.2204.
- [5] M. Summerfield, *Rapid GUI Programming with Python and Qt: The Definitive Guide to PyQt Programming*, 1st ed. USA: Prentice Hall Press, 2015.
- [6] Y. A. Akbari, *Python Programming : Black Book*. 2020.
- [7] M. Inden, “Short Introduction to pytest,” in *Python Challenges: 100 Proven Programming Tasks Designed to Prepare You for Anything*, Berkeley, CA: Apress, 2022, pp. 615–622. doi: 10.1007/978-1-4842-7398-2_11.
- [8] B. Okken, *Python Testing with pytest*. Pragmatic Bookshelf, 2022.
- [9] A. Ronacher, “Jinja2 documentation,” *Welcome to Jinja2—Jinja2 Documentation (2.8-dev)*, 2008.
- [10] M. E. Khan and F. Khan, “A Comparative Study of White Box, Black Box and Grey Box Testing Techniques,” 2012. [Online]. Available: www.ijacsa.thesai.org
- [11] T. Kuruppu *et al.*, “Source Code based Approaches to Automate Marking in Programming Assignments,” Apr. 2021, pp. 291–298. doi: 10.5220/0010400502910298.
- [12] M. Striwe and M. Goedicke, “A Review of Static Analysis Approaches for Programming Exercises,” in *Communications in Computer and Information Science*, 2014, vol. 439, pp. 100–113. doi: 10.1007/978-3-319-08657-6_10.
- [13] T. Wang, X. Su, Y. Wang, and P. Ma, “Semantic similarity-based grading of student programs,” *Information and Software Technology*, vol. 49, no. 2, pp. 99–107, Feb. 2007, doi: 10.1016/j.infsof.2006.03.001.



- [14] AltexSoft, “Functional and Nonfunctional Requirements: Specification and Types,” 2021.
<https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/> (accessed Jul. 09, 2022).
- [15] ReQtest, “How Non-functional Requirements add value to software development?,” 2011.
<https://reqtest.com/requirements-blog/what-are-non-functional-requirements/> (accessed Jul. 09, 2022).
- [16] G. Hagerer, L. Lahesoo, M. Anschütz, S. Krusche, and G. Groh, “An Analysis of Programming Course Evaluations Before and After the Introduction of an Autograder.” Jul. 2021.