

DAFTAR PUSTAKA

- [1] T. Sedano, “Code Readability Testing, an Empirical Study.”
- [2] R. Baggen, · José, P. Correia, K. Schill, J. Visser, and J. P. Correia, “Standardized Code Quality Benchmarking for Improving Software Maintainability.”
- [3] M. Majthoub, M. H. Qutqut, and Y. Odeh, “Software Re-engineering: An Overview.”
- [4] R. S. Pressman, *Software Engineering: A Practitioner’s Approach*.
- [5] A. T. Sondha, U. Sa’adah, F. F. Hardiansyah, M. Bagus, and A. Rasyid, “Framework dan Code Generator Pengembangan Aplikasi Android dengan Menerapkan Prinsip Clean Architecture,” 2020.
- [6] “ViewModel Overview | Android Developers.”
<https://developer.android.com/topic/libraries/architecture/viewmodel>
(accessed Feb. 06, 2022).
- [7] J. Wharton, “JakeWharton/butterknife: Bind Android views and callbacks to fields and methods.,” Aug. 13, 2020.
<https://github.com/JakeWharton/butterknife> (accessed Feb. 06, 2022).
- [8] R. Kumar, *Research Methodology: A Step-by-Step Guide for Beginners*, 4th ed. SAGE Publicaitions, 2015.
- [9] R. Miles and K. Hamilton, *Learning UML 2.0*. Newton: O'Reilly Media, Inc., 2006.
- [10] R. C. Martin, *Clean Architecture A Craftman’s Guide to Software Structure and Design*.
- [11] C. Jessica, “Point of Sales: Pengertian, Jenis, dan Fungsinya Bagi Bisnis - Glints Blog.” <https://glints.com/id/lowongan/apa-itu-point-of-sales/>
(accessed Feb. 17, 2022).
- [12] J. Fernandes Andry, R. E. Riwanto, R. L. Wijaya, A. A. Prawoto, and T. Prayogo, “Development Point of Sales Using SCRUM Framework,” 2019, doi: 10.20470/jsi.v10i1.359.
- [13] C. S. Brown, “What is Android? Everything you need to know about Google’s OS,” *Android Authority*, 2021.

- <https://www.androidauthority.com/what-is-android-328076/> (accessed Feb. 17, 2022).
- [14] Putra, “PENGERTIAN ANDROID: Sejarah, Kelebihan & Versi Sistem Operasi,” *Salamandian*, 2019. <https://salamadian.com/pengertian-android/> (accessed Feb. 17, 2022).
- [15] J. Raphael, “Android versions: A living history from 1.0 to 12 | Computerworld,” *COMPUTERWORLD*, 2021. <https://www.computerworld.com/article/3235946/android-versions-a-living-history-from-1-0-to-today.html> (accessed Feb. 17, 2022).
- [16] “Java Tutorial | Learn Java Programming - javatpoint,” *Javaatpoint*. <https://www.javatpoint.com/java-tutorial> (accessed Feb. 17, 2022).
- [17] S. Bose Student, M. Mukherjee Student, and M. Banerjee Asst, “A COMPARATIVE STUDY: JAVA VS KOTLIN PROGRAMMING IN ANDROID APPLICATION DEVELOPMENT,” *International Journal of Advanced Research in Computer Science*, vol. 9, no. 3, doi: 10.26483/ijarcs.v9i3.5978.
- [18] L. M. Laird and M. C. Brennan, “Software Measurement and Estimation A Practical Approach”.
- [19] T. J. McCabe, “A Complexity Measure,” *IEEE Transactions on Software Engineering*, vol. SE-2, no. 4, pp. 308–320, 1976, doi: 10.1109/TSE.1976.233837.
- [20] “Cyclomatic Complexity Defined || FREE Demo || Video Explanation.” <https://www.castsoftware.com/glossary/cyclomatic-complexity> (accessed Feb. 17, 2022).
- [21] A. H. Sa'diyyah, “Eliminasi findViewById dengan ButterKnife Library,” *Codepolitan*, 2017. <https://www.codepolitan.com/eliminasi-findviewbyid-dengan-butterknife-library-5a44ce9421bab> (accessed Feb. 18, 2022).
- [22] A. C. Graciamary, “Enhanced Re-Engineering Mechanism to Improve the Efficiency of Software Re-Engineering,” *IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 7, no. 11, 2016, [Online]. Available: www.ijacsa.thesai.org

- [23] A. C. Graciamary and M. Chidambaram, “EESRM: An Effective Approach to Improve the Performance of Software Re-Engineering,” *International Journal of Applied Engineering Research*, vol. 13, no. 6, pp. 3648–3654, 2018, Accessed: Feb. 18, 2022. [Online]. Available: <http://www.rippublication.com>
- [24] B. Dorninger, M. Moser, and J. Pichler, “Multi-language re-documentation to support a COBOL to Java migration project,” *SANER 2017 - 24th IEEE International Conference on Software Analysis, Evolution, and Reengineering*, pp. 536–540, Mar. 2017, doi: 10.1109/SANER.2017.7884669.
- [25] A. C. Beltrão, F. de Almeida Farzat, and G. H. Travassos, “Technical Debt: A Clean Architecture Implementation”, [Online]. Available: <https://www.sonarqube.org>
- [26] M. Pavlova, “Design Patterns,” *Sourcemaking*, 2022. https://sourcemaking.com/design_patterns (accessed Feb. 18, 2022).
- [27] B. Joshi, “Overview of SOLID Principles and Design Patterns,” *Beginning SOLID Principles and Design Patterns for ASP.NET Developers*, pp. 1–44, 2016, doi: 10.1007/978-1-4842-1848-8_1.
- [28] A. Mukharil Bachtiar, D. Dharmayanti, and M. K. Sabariah, “ANALISIS KUALITAS PERANGKAT LUNAK TERHADAP SISTEM INFORMASI UNIKOM,” *Majalah Ilmiah UNIKOM*, vol. 11, no. 2.
- [29] B. R. Reddy and A. Ojha, “Performance of Maintainability Index prediction models: a feature selection based study,” *Evolving Systems*, vol. 10, no. 2, pp. 179–204, Jun. 2019, doi: 10.1007/S12530-017-9201-0/TABLES/10.
- [30] R. Ganang Atmaja, B. Priyambadha, and F. Pradana, “Pembangunan Kakas Bantu Untuk Mengukur Maintainability Index Pada Perangkat Lunak Berdasarkan Nilai Halstead Metrics dan McCabe’s Cyclomatic Complexity,” vol. 3, no. 3, pp. 2167–2172, 2019, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [31] M. Flower, *Refactoring: Improving the Design of Existing Code*, 2nd Edition. Addison-Wesley Professional, 2018.

- [32] R. C. Martin, *Clean Code: A Handbook of Agile Software Craftsmanship*. Boston: Pearson Education, 2009.
- [33] J. Hartono, *Analisis dan Desain*. Yogyakarta, 2005.
- [34] “MVVM Model View ViewModel Part - 1 - TechNet Articles - United States (English) - TechNet Wiki.” <https://social.technet.microsoft.com/wiki/contents/articles/13347.mvvm-model-view-viewmodel-part-1.aspx> (accessed Feb. 18, 2022).
- [35] T. Lou, “A comparison of Android Native App Architecture MVC, MVP and MVVM.”
- [36] “Mengenal Android Studio | Developer Android | Android Developers.” <https://developer.android.com/studio/intro?hl=ID> (accessed Feb. 18, 2022).
- [37] “Software Engineering | Integration Testing,” *GeeksforGeeks*, Jun. 13, 2022. <https://www.geeksforgeeks.org/software-engineering-integration-testing/> (accessed Aug. 14, 2022).
- [38] “Integration Testing,” *SOFTWARE TESTING Fundamentals*, Sep. 13, 2020. <https://softwaretestingfundamentals.com/integration-testing/> (accessed Aug. 14, 2022).
- [39] A. R. Shaikh, “Applying Halstead Metrics in Your Programs”.
- [40] “Dependency injection with Hilt | Android Developers.” <https://developer.android.com/training/dependency-injection/hilt-android> (accessed Jun. 16, 2022).
- [41] R. Singh and I. A. Khan, “AN APPROACH FOR INTEGRATION TESTING IN ONLINE RETAIL APPLICATIONS,” *International Journal of Computer Science & Information Technology (IJCSIT)*, vol. 4, no. 3, 2012, doi: 10.5121/ijcsit.2012.4312.
- [42] “Guide to Android app modularization | Android Developers.” <https://developer.android.com/topic/modularization> (accessed Aug. 14, 2022).