1. <https://www.androidpolice.com/2019/05/07/there-are-now-more-than-2-5-billion-active-android-devices/> [Online]. [Accessed: 25.01.2020].
2. Wasserman 2010
3. <https://www.peerbits.com/blog/update-mobile-app.html> [Online]. [Accessed: 15.02.2020].
4. Jussi Koskinen. Software maintenance costs. Jyvaskyl ¨ a: University of Jyv ¨ askyl ¨ a¨, 2010.
5. <https://source.android.com/> [Online]. [Accessed: 05.04.2020].
6. <http://www.openhandsetalliance.com/oha_members.html> [Online]. [Accessed: 05.04.2020].
7. <https://developer.android.com/guide/platform>[Online]. [Accessed: 12.04.2020].
8. <https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/> [Online]. [Accessed: 13.04.2020].
9. [https://developer.android.com/guide/components](https://developer.android.com/guide/components/fundamentals) [Online]. [Accessed: 14.04.2020]

# Clean Architecture: A Craftsman's Guide to Software Structure and Design, First Edition 2018 Pearson Education, Inc.

1. Robert C Martin. Clean code: a handbook of agile software craftsmanship. Pearson Education, 2009.
2. Igor Bugayenko. Elegant Objects. Createspace Independent Publishing Platform
3. J. R. Mckee, “Maintenance as a Function of Design”. Proceedings AFIPS, National Computer Conference, Las Vegas, pp 187-93.
4. Guidelines for Architecting Android Apps: A Mixed-Method Empirical Study
5. F.P. Brooks. The mythical man-month: essays on software engineering. Addison-Wesley Pub. Co., 1975.
6. <https://martinfowler.com/architecture/> [Online] [Accessed: 16.04.2020].
7. Philippe Kruchten, *The Rational Unified Process: An Introduction*, Third Edition. Addison-Wesley Professional 2003.
8. KÄLLSTRÖM, Hugo. Increasing Maintainability for Android Applications: Implementation and Evaluation of Three Software Architectures on the Android Framework. 2016R.
9. A Tale of Two Development Approach: Empirical Study on The Maintainability and Modularity of Android Mobile Application with Anti-Pattern and Model-View-Presenter Design Pattern
10. [IEEE Standard Glossary of Software Engineering Terminology](http://standards.ieee.org/reading/ieee/std_public/description/se/610.12-1990_desc.html)
11. [Refactoring: Improving the Design of Existing Code](https://www.amazon.co.uk/Refactoring-Improving-Design-Existing-Technology/dp/0201485672) 1999 Martin fowler
12. C. Jones, "The economics of software maintenance in the twenty first century," Unpublished manuscript. http://citeseerx. ist. psu. edu/viewdoc/summary, 2006
13. Application of Ensemble Techniques in Predicting Object-Oriented Software Maintainability Hadeel Alsolai
14. Foundations for the Study of Software Architecture
15. Software Architecture: a Roadmap David Garlan
16. Software Metrics Proposal for Conformity Checking of Class Diagram to SOLID Design Principles
17. Separation of Concerns Walter L. Hursch and Cristina Videira Lopes
18. Degrees of Separation: Multi-Dimensional Separation of Concerns
19. <https://developer.android.com/kotlin> [Online] [Accessed: 18.03.2021].
20. <https://developer.android.com/reference/android/os/AsyncTask> [Online] [Accessed: 18.03.2021]
21. <https://developer.android.com/topic/libraries/architecture/coroutines> [Online] [Accessed: 18.03.2021]
22. <https://developer.android.com/training/dependency-injection/hilt-android> [Online] [Accessed: 18.03.2021].
23. Review on ‘Maintainability’ Metrics in Open Source Software, Abubakar Diwani Bakar
24. Evaluating Maintainability of Android Applications, Saifan
25. T. J. McCabe, “A complexity measure,” IEEE Transactions on software Engineering, no. 4, pp. 308–320, 1976.
26. C. Chidamber, Towards a Metrics Suite for Object Oriented Design, ACM, OOPSL’91, pp. 197-211, 1991.
27. <https://www.codemr.co.uk> [Online] [Accessed: 21.03.2021].
28. Experiences on Managing Technical Debt with Code Smells and AntiPatterns ramirez
29. Reversing Entropy in a Software Development Project: Technical Debt and Anti Patterns