СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Дубовик А. П. Генерація тестів GUI Android-додатків: аналіз сучасних підходів та виклики декларативної парадигми. Інженерія програмного забезпечення і передові інформаційні технології (Soft Tech-2023): матеріали V Міжнародної наук.-практ. конф. молодих вчених та студентів, м. Київ, 19-21 грудня 2023 року, НТУУ «Київський політехнічний інститут імені Ігоря Сікорського», 2023. С. 108-113.

2. Дубовик А. П. Особливості генерування коду модульних тестів GUI Android-додатків, розроблених з використанням декларативної парадигми (Soft Tech-2024): матеріали VI Міжнародної наук.-практ. конф. молодих вчених та студентів, м. Київ, 20-21 травня 2024 року, НТУУ «Київський політехнічний інститут імені Ігоря Сікорського», 2024.

3. Jetpack Compose UI – App Development Toolkit. Android Official Documentation. [Електронний ресурс] // Режим доступу: https://developer.android.com/jetpack/compose (дата звернення: 01.12.2023)

4. Catalin Ghita, Kickstart Modern Android Development with Jetpack and Kotlin: Enhance your applications by integrating Jetpack and applying modern app architectural concepts, Packt Publishing, 2022.

5. Google, "Develop android apps with kotlin", [Електронний ресурс] // Режим доступу:: https://developer.android.com/kotlin, 2020, (дата звернення: 01.12.2023).

6. GitHub, "The state of the octoverse: Top languages", 2019, [Електронний ресурс] // Режим доступу: https://octoverse.github.com/#top-languages (дата звернення: 01.12.2023).

7. D. Winer, "Android’s commitment to kotlin", [Електронний ресурс] // Режим доступу: https://android-developers.googleblog.com/2019/12/androids-commitment-to-kotlin.html (дата звернення: 15.12.2023).

8. AppBrain, "Android statistics / android libraries / kotlin", [Електронний ресурс] // Режим доступу: https://www.appbrain.com/stats/libraries/details/kotlin

/kotlin (дата звернення: 15.12.2023)

9. G. Hecht and A. Bergel, "Quantifying the adoption of Kotlin on Android stores: Insight from the bytecode", 2021 IEEE/ACM 8th International Conference on Mobile Software Engineering and Systems (MobileSoft), Madrid, Spain, 2021, pp. 94-98, doi: 10.1109/MobileSoft52590.2021.00019.

10. JUnit5: Official Documentation [Електронний ресурс]. Режим доступу: https://junit.org/junit5/.

11. Mockito framework site [Електронний ресурс]. Режим доступу: https://site.mockito.org/.

12. Robolectric: Official Documentation [Електронний ресурс]. Режим доступу: https://robolectric.org/.

13. Appium: Official Documentation [Електронний ресурс]. Режим доступу: https://appium.io/docs/en/latest/.

14. Espresso: Official Android Documentation [Електронний ресурс]. Режим доступу: https://developer.android.com/training/testing/espresso.

15. UI Automator: Official Android Documentation [Електронний ресурс]. Режим доступу: https://developer.android.com/training/testing/other-components/ui-automator.

16. Calabash: Automated Functional testing for Android using cucumber [Електронний ресурс]. Режим доступу: https://github.com/calabash/calabash-android.

17. MonkeyRunner: Official Android Documentation [Електронний ресурс]. Режим доступу: https://developer.android.com/studio/test/monkeyrunner

18. Robotium: Android UI Testing [Електронний ресурс]. Режим доступу: https://github.com/RobotiumTech/robotium.

19. Beizer B. Software Testing Techniques / Boris Beizer., 1990. – 580 p. – (2nd Edition).

20. Mike Cohn. Succeeding with Agile. Software Development Using Scrum: // Mike Cohn. Addison-Wesley Professional. 2010. – 512 p.

21. Fields J. Working Effectively with Unit Tests // Jay Fields. Leanpub.

2015. – 347 p.

22. Kong, Pingfan, Li Li, Jun Gao, Kui Liu, Tegawendé F. Bissyandé and Jacques Klein. “Automated Testing of Android Apps: A Systematic Literature Review.” IEEE Transactions on Reliability 68 (2019): pp. 45-66.

23. Nie, Liming, Kabir S. Said, Lingfei Ma, Yaowen Zheng and Yangyang Zhao. “A systematic mapping study for graphical user interface testing on mobile apps.” IET Softw. 17 (2023): pp. 249-267.

24. Samir, Amira & Amin, Huda & Badr, Nagwa. (2022). A survey on automated user interface testing for mobile applications. [Електронний ресурс]: pp. 126-136. 10.21608/ijicis.2022.98138.1124.

25. Usman, Asmau & Ibrahim, Noraini & Anka, Salihu. (2020). TEGDroid: Test Case Generation Approach for Android Apps Considering Context and GUI Events. International Journal on Advanced Science, Engineering and Information Technology [Електронний ресурс]. 10. 16. 10.18517/ijaseit.10.1.10194.

26. Machiry, Aravind & Tahiliani, Rohan & Naik, Mayur. (2013) [Електронний ресурс]: Dynodroid: An input generation system for Android apps. pp. 224-234. 10.1145/2491411.2491450.

27. J. Doyle, T. Saber, P. Arcaini and A. Ventresque, "Improving Mobile User Interface Testing with Model Driven Monkey Search" [Електронний ресурс]: 2021 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), Porto de Galinhas, Brazil, 2021, pp. 138-145, doi: 10.1109/ICSTW52544.2021.00034.

28. Köroğlu, Yavuz & Sen, Alper. (2020) [Електронний ресурс]: Functional test generation from UI test scenarios using reinforcement learning for android applications. Software Testing, Verification and Reliability. pp. 31. 10.1002/stvr.1752.

29. Yasin, Husam & Ab hamid, Siti hafizah & Raja Yusof, Raja. (2021) [Електронний ресурс]: DroidbotX: Test Case Generation Tool for Android Applications Using Q-Learning. Symmetry. pp. 13. 310. 10.3390/sym13020310.

30. Kabir S. Said, Liming Nie, Adekunle A. Ajibode, and Xueyi Zhou. 2021 [Електронний ресурс]: GUI testing for mobile applications: objectives, approaches

and challenges. In Proceedings of the 12th Asia-Pacific Symposium on Internetware (Internetware '20). Association for Computing Machinery, New York, NY, USA, pp. 51–60. https://doi.org/10.1145/3457913.3457931

31. Mariani, L.; Pezze, M.; Riganelli, O.; Santoro, M. Autoblacktest: Automatic black-box testing of interactive applications. In Proceedings of the IEEE Fifth International Conference on Software Testing, Verification and Validation, Montreal, QC, Canada, 17–21 April 2012; pp. 81–90.

32. Esparcia-Alcázar, A.I.; Almenar, F.; Martínez, M.; Rueda, U.; Vos, T. Q-learning strategies for action selection in the TESTAR automated testing tool. In Proceedings of the 6th International Conferenrence on Metaheuristics and Nature Inspired Computing (META 2016), Marrakech, Morocco, 27–31 October 2016; pp. 130–137.

33. Kim, J.; Kwon, M.; Yoo, S. Generating test input with deep reinforcement learning. In Proceedings of the IEEE/ACM 11th International Workshop on Search-Based Software Testing (SBST), Gothenburg, Sweden, 28–29 May 2018; pp. 51–58.

34. Koroglu, Y.; Sen, A.; Muslu, O.; Mete, Y.; Ulker, C.; Tanriverdi, T.; Donmez, Y. QBE: QLearning-based exploration of android applications. In Proceedings of the IEEE 11th International Conference on Software Testing, Verification and Validation (ICST), Luxembourg, 18–22 March 2013; pp. 105–115.

35. Vuong, T.A.T.; Takada, S. A reinforcement learning based approach to automated testing of Android applications. In Proceedings of the 9th ACM SIGSOFT International Workshop on Automating TEST Case Design, Selection, and Evaluation, Lake Buena Vista, FL, USA, 5 November 2018; pp. 31–37.

36. Adamo, D.; Khan, M.K.; Koppula, S.; Bryce, R. Reinforcement learning for Android GUI testing. In Proceedings of the 9th ACM SIGSOFT International Workshop on Automating TEST Case Design, Selection, and Evaluation, Lake Buena Vista, FL, USA, 5 November 2018; pp. 2–8.

37. Gu, T.; Cao, C.; Liu, T.; Sun, C.; Deng, J.; Ma, X.; Lü, J. Aimdroid: Activity-insulated multi-level automated testing for android applications. In Proceedings of the IEEE International Conference on Software Maintenance and

Evolution (ICSME), Shanghai, China, 17–22 September 2017; pp. 103–114.

38. Li, Y.; Yang, Z.; Guo, Y.; Chen, X. A Deep Learning based Approach to Automated Android App Testing. arXiv 2019, arXiv:1901.02633.

39. Li, Y.; Yang, Z.; Guo, Y.; Chen, X. DroidBot: A lightweight UI-guided test input generator for Android. In Proceedings of the IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C), Buenos Aires, Argentina, 20–28 May 2017; pp. 23–26.

40. Chen, T.Y.; Kuo, F.-C.; Merkel, R.G.; Tse, T. Adaptive random testing: The art of test case diversity. J. Syst. Softw. 2010, 83, pp. 60–66.

41. Google. UI/Application Exerciser Monkey|Android Developers. [Електронний ресурс] // Режим доступу: https://developer.android.com/studio/test /monkey.

42. Wang, W.; Li, D.; Yang, W.; Cao, Y.; Zhang, Z.; Deng, Y.; Xie, T. An empirical study of android test generation tools in industrial cases. In Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering, Montpellier, France, 3–7 September 2018; pp. 738–748.

43. Mahmood, R.; Mirzaei, N.; Malek, S. Evodroid: Segmented evolutionary testing of android apps. In Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering, Hong Kong, China, 11 November 2014; pp. 599–609.

44. Clapp, L.; Bastani, O.; Anand, S.; Aiken, A. Minimizing GUI event traces. In Proceedings of the 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering, Seattle, WA, USA, 13–18 November 2016; pp. 422–434.

45. Zheng, H.; Li, D.; Liang, B.; Zeng, X.; Zheng, W.; Deng, Y.; Lam, W.; Yang, W.; Xie, T. Automated test input generation for android: Towards getting there in an industrial case. In Proceedings of the IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP), Buenos Aires, Argentina, 20–28 May 2017; pp. 253–262.

46. Hu, C.; Neamtiu, I. Automating GUI testing for Android applications. In Proceedings of the 6th International Workshop on Automation of Software Test,

Waikiki, Honolulu, HI, USA, 23–24 May 2011; pp. 77–83.

47. Machiry, A.; Tahiliani, R.; Naik, M. Dynodroid: An input generation system for android apps. In Proceedings of the 9th Joint Meeting on Foundations of Software Engineering, Saint Petersburg, Russia, 18–26 August 2013; pp. 224–234.

48. Haoyin, L. Automatic android application GUI testing—A random walk approach. In Proceedings of the International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 22–24 March 2017; pp. 72–76.

49. Azim, T.; Neamtiu, I. Targeted and depth-first exploration for systematic testing of android apps. In Proceedings of the ACM SIGPLAN International Conference on Object Oriented Programming Systems Languages & Applications, Indianapolis, IN, USA, 26–31 October 2013; pp. 641–660.

50. Su, T.; Meng, G.; Chen, Y.; Wu, K.; Yang, W.; Yao, Y.; Pu, G.; Liu, Y.; Su, Z. Guided, stochastic model-based GUI testing of Android apps. In Proceedings of the 11th Joint Meeting on Foundations of Software Engineering, Paderborn, Germany, 4–8 September 2017; pp. 245–256.

51. Baek, Y.-M.; Bae, D.-H. Automated model-based Android GUI testing using multi-level GUI comparison criteria. In Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering, Singapore, 3–7 September 2016; pp. 238–249.

52. Yang, W.; Prasad, M.R.; Xie, T. A grey-box approach for automated GUI-model generation of mobile applications. In Proceedings of the International Conference on Fundamental Approaches to Software Engineering, Rome, Italy, 16–24 March; pp. 250–265.

53. Hao, S.; Liu, B.; Nath, S.; Halfond, W.G.; Govindan, R. PUMA: Programmable UI-automation for large-scale dynamic analysis of mobile apps. In Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services, Bretton Woods, NH, USA, 16–19 June 2014; pp. 204–217.

54. Gu, T.; Sun, C.; Ma, X.; Cao, C.; Xu, C.; Yao, Y.; Zhang, Q.; Lu, J.; Su, Z. Practical GUI testing of Android applications via model abstraction and refinement.

In Proceedings of the IEEE/ACM 41st International Conference on Software Engineering (ICSE), Montreal, QC, Canada, 25–31 May 2019; pp. 269–280.

55. Adamsen, C.Q.; Mezzetti, G.; Møller, A. Systematic execution of android test suites in adverse conditions. In Proceedings of the International Symposium on Software Testing and Analysis, Baltimore, MD, USA, 13–17 July 2015; pp. 83–93.

56. Moran, K.; Linares-Vásquez, M.; Bernal-Cárdenas, C.; Vendome, C.; Poshyvanyk, D. Automatically discovering, reporting and reproducing android application crashes. In Proceedings of the IEEE International Conference on Software Testing, Verification And Validation (ICST), Chicago, IL, USA, 11–15 April 2016; pp. 33–44.

57. Hu, G.; Yuan, X.; Tang, Y.; Yang, J. Efficiently, effectively detecting mobile app bugs with appdoctor. In Proceedings of the Ninth European Conference on Computer Systems, Amsterdam, The Netherlands, 14–16 April 2014; p. 18.

58. Mirzaei, N.; Bagheri, H.; Mahmood, R.; Malek, S. Sig-droid: Automated system input generation for android applications. In Proceedings of the IEEE 26th International Symposium on Software Reliability Engineering (ISSRE), Gaithersbury, MD, USA, 2–5 November 2015; pp. 461–471.

59. Mao, K.; Harman, M.; Jia, Y. Sapienz: Multi-objective automated testing for Android applications. In Proceedings of the 25th International Symposium on Software Testing and Analysis, Saarbrücken, Germany, 18–20 July 2016; pp. 94–105.

60. Yasin, Husam N., Siti Hafizah Ab Hamid, and Raja Jamilah Raja Yusof. 2021. "DroidbotX: Test Case Generation Tool for Android Applications Using Q-Learning" Symmetry 13, no. 2: 310. https://doi.org/10.3390/sym13020310

61. Symflower for IntelliJ IDEA - Smart Unit Test Generator for Java. Official Website. [Електронний ресурс] // Режим доступу: https://symflower.com/en/ (дата звернення: 12.02.2024)

62. AI-Powered Java Testing Tool - Boost Productivity, Official Website. [Електронний ресурс] // Режим доступу: https://www.parasoft.com/products/para soft -jtest/ (дата звернення: 13.02.2024)

63. EvoSuite | Automatic Test Suite Generation for Java, Official Website.

[Електронний ресурс] // Режим доступу: https://www.evosuite.org/ (дата звернення: 15.02.2024)

64. Google, Thinking in Compose. Official Documentation. [Електронний ресурс] // Режим доступу: https://developer.android.com/jetpack/compose/mental-model (дата звернення: 18.02.2024)

65. Google, Compose modifiers. Official Documentation. [Електронний ресурс] // Режим доступу: https://developer.android.com/jetpack/compose/modifiers (дата звернення: 20.02.2024)

66. Google, Testing your Compose layout. Official Documentation. [Електронний ресурс] // Режим доступу: https://developer.android.com/jetpack/compose/testing (дата звернення: 22.02.2024)

67. Google, Android’s Kotlin-first approach. Official Documentation. [Електронний ресурс] // Режим доступу: https://developer.android.com/kotlin/first (дата звернення: 22.02.2024)

68. Kotlin for Android. Official Documentation. [Електронний ресурс] // Режим доступу: https://kotlinlang.org/docs/android-overview.html (дата звернення: 22.02.2024)