Object-oriented Software Documentation

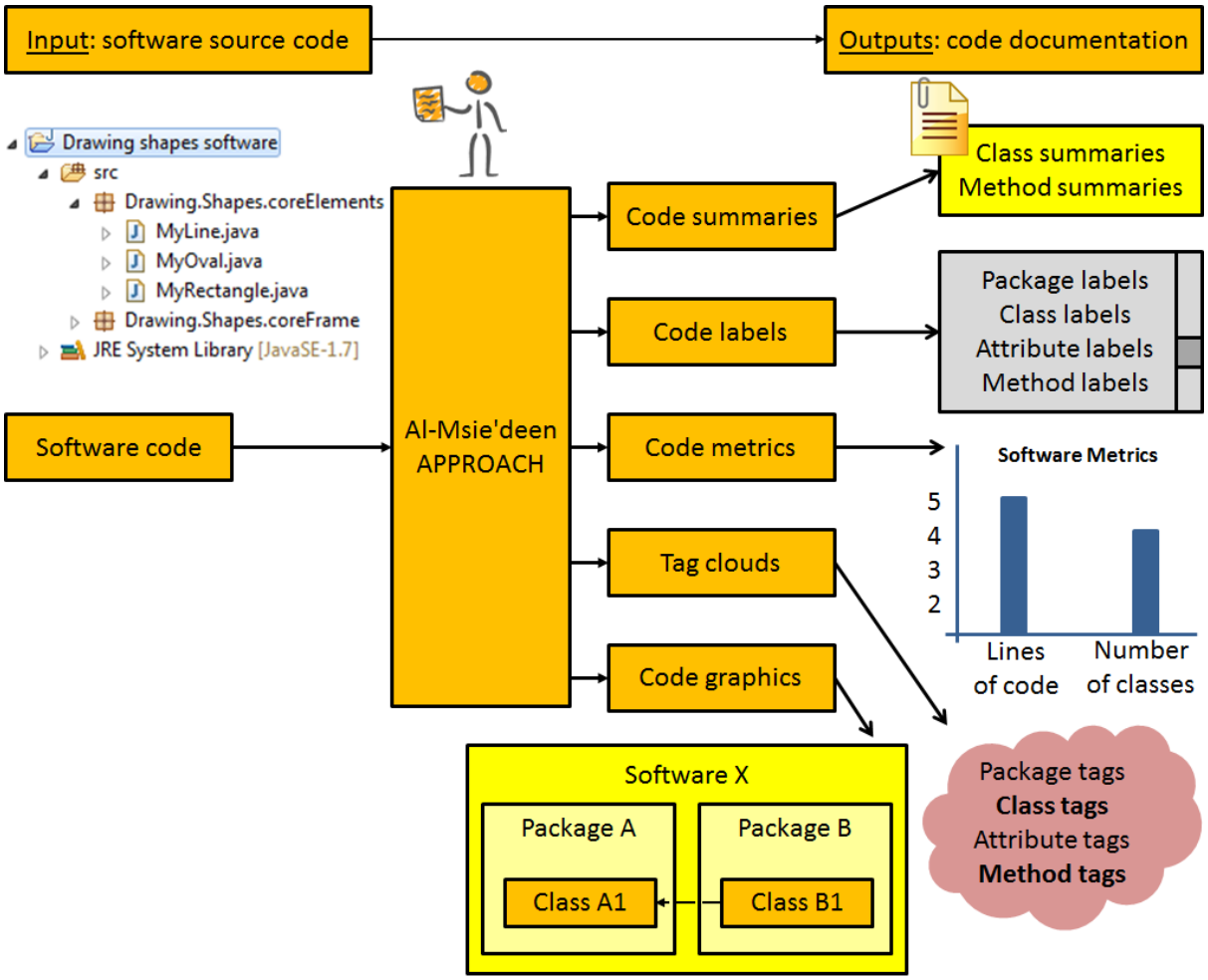
Ra'Fat Al-Msie'deen

2019

Abstract—Software documentation provides software developers and users with a useful description of what the software does, how it works, how it's built, and how it's supposed to be used. For instance, a user manual demonstrates how users should interact with a software product; while the design document helps developers during software evolution and maintenance. The main challenge is to construct automated tool that can produce, on-demand, a dynamic documentation of software source code. This book presents an automatic approach called Al-Msie'deen approach to generate on-demand software documentation for software code. The novelty of this approach lies in the production of dynamic software code documentation at package, class, attribute, and method levels. Code documentation is a written text or illustrated diagrams that present information about all levels of code granularity, from packages level down to methods level. Al-Msie'deen approach generates different kinds of code documentation such as code summaries, labels, metrics, tag clouds, and graphics documents. Code documents make information on software code more accessible to developers and cheaper to create. This book presents the results of experiments conducted on Drawing shapes software, Rhino, NanoXML, Mobile media and ArgoUML software systems.

Keywords—Software engineering, software documentation, software visualization, software comprehension, object-oriented software, source code summarization, code labels, software metrics, tag clouds.

Department of Computer Information Systems, Faculty of IT
Mutah University, Karak, Jordan
rafatalmsiedeen@mutah.edu.jo


Input: software source code
Outputs: code documentation
Drawing shapes software
src
Drawing.Shapes.coreElements
MyLine.java
MyOval.java
MyRectangle.java
Drawing.Shapes.coreFrame
JRE System Library [JavaSE-1.7]
Software code
Al-Msie'deen APPROACH
Code summaries
Code labels
Code metrics
Tag clouds
Code graphics
Class summaries
Method summaries
Package labels
Class labels
Attribute labels
Method labels
Software Metrics
5
4
3
2
Lines of code
Number of classes
Software X
Package A
Package B
Class A1
Class B1
Package tags
Class tags
Attribute tags
Method tags

REFERENCES

[1] R. Al-Msie'deen, "Visualizing object-oriented software for understanding and documentation," International Journal of Computer Science and Information Security, vol. 13, no. 5, pp. 18–27, 2015.

[2] A. Forward and T. Lethbridge, "The relevance of software documentation, tools and technologies: a survey," in Proceedings of the 2002 ACM Symposium on Document Engineering, McLean, Virginia, USA, November 8-9, 2002. ACM, 2002, pp. 26–33.

[3] P. Caserta and O. Zendra, "Visualization of the static aspects of software: A survey," IEEE Trans. Vis. Comput. Graph., vol. 17, no. 7, pp. 913–933, 2011.

[4] P. W. McBurney and C. McMillan, "Automatic documentation generation via source code summarization of method context," in 22nd International Conference on Program Comprehension, ICPC 2014, Hyderabad, India, June 2-3, 2014. ACM, 2014, pp. 279–290.

[5] C. A. M. Grant, "Software visualization in prolog," Ph.D. dissertation, University of Cambridge, UK, 2000.

[6] S. Diehl, Software Visualization - Visualizing the Structure, Behaviour, and Evolution of Software. Springer, 2007.

[7] H. A. Mu¨ller, S. R. Tilley, and K. Wong, "Understanding software systems using reverse engineering technology perspectives from the Rigi project," in Proceedings of the 1993 Conference of the Centre for Advanced Studies on Collaborative Research, October 24-28, 1993, Toronto, Ontario, Canada, 2 Volumes. IBM, 1993, pp. 217–226.

[8] V. Rajlich and N. Wilde, "The role of concepts in program comprehension," in 10th International Workshop on Program Comprehension (IWPC 2002), 27-29 June 2002, Paris, France. IEEE Computer Society, 2002, pp. 271–278.

[9] N. Pereira, M. J. V. Pereira, and P. R. Henriques, "Comment-based concept location over system dependency graphs," in 3rd Symposium on Languages, Applications and Technologies, SLATE 2014, June 19-20, 2014 - Braganc¸a, Portugal. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2014, pp. 51–58.

[10] E. J. Chikofsky and J. H. C. II, "Reverse engineering and design recovery: A taxonomy," IEEE Software, vol. 7, no. 1, pp. 13–17, 1990.

[11] K. H. Bennett and V. Rajlich, "Software maintenance and evolution: a roadmap," in

22nd International Conference on Software Engineering, Future of Software Engineering Track, ICSE 2000, Limerick Ireland, June 4-11, 2000. ACM, 2000, pp. 73–87.

[12] N. H. Madhavji, J. Fernandez-Ramil, and D. Perry, Software Evolution and Feedback: Theory and Practice. USA: John Wiley and Sons, Inc., 2006.

[13] C. W. Krueger, "Software reuse," ACM Comput. Surv., vol. 24, no. 2, pp. 131–183, Jun. 1992.

[14] J. S. Poulin, J. M. Caruso, and D. R. Hancock, "The business case for software reuse," IBM Systems Journal, vol. 32, no. 4, pp. 567–594, 1993.

[15] S. A. Bohner, "Impact analysis in the software change process: a year 2000 perspective," in 1996 International Conference on Software Maintenance (ICSM '96), 4-8 November 1996, Monterey, CA, USA, Proceedings. IEEE Computer Society, 1996, pp. 42–51.

[16] R. Koschke, "Software visualization in software maintenance, reverse engineering, and re-engineering: a research survey," Journal of Software Maintenance, vol. 15, no. 2, pp. 87–109, 2003.

[17] A. Al-Shamailh, R. Al-Msie'deen, and A. Alsarhan, "Comparison between the rules of data storage tools," International Journal of Database Theory and Application, vol. 8, no. 1, pp. 129–136, 2015.

[18] J. I. Maletic, A. Marcus, G. Dunlap, and J. Leigh, "Visualizing object-oriented software in virtual reality," in 9th International Workshop on Program Comprehension (IWPC 2001), 12-13 May 2001, Toronto, Canada. IEEE Computer Society, 2001, pp. 26–35.

[19] S. Tichelaar, S. Ducasse, and S. Demeyer, "FAMIX: exchange experiences with CDIF and XMI," in Proceedings of the ICSE 2000 Workshop on Standard Exchange Format (WoSEF 2000), 2000, pp. 1–7.

[20] S. Tichelaar, S. Ducasse, S. Demeyer, and O. Nierstrasz, "A meta-model for language-independent refactoring," in Proceedings International Symposium on Principles of Software Evolution, Nov 2000, pp. 154–164.

[21] R. Al-Msie'deen. Prototypes. https://sites.google.com/site/ralmsideen/tools. Accessed: 2019-11-10.

[22] Y. Smaragdakis and C. Csallner, "Combining static and dynamic reasoning for bug detection," in Tests and Proofs, First International Conference, TAP 2007, Zurich, Switzerland, February 12-13, 2007. Revised Papers. Springer, 2007, pp. 1–16.

[23] M. Egele, T. Scholte, E. Kirda, and C. Kruegel, "A survey on automated dynamic

malware-analysis techniques and tools," ACM Comput. Surv., vol. 44, no. 2, pp. 6:1–6:42, 2012.

[24] J. L. Overbey, "Immutable source-mapped abstract syntax tree: A design pattern for refactoring engine APIs," in Proceedings of the 20th Conference on Pattern Languages of Programs, ser. PLoP'13. USA: The Hillside Group, 2013, pp. 7:1–7:8.

[25] G. Rakic and Z. Budimac, "Introducing enriched concrete syntax trees," in Proceedings of the 14th International Multiconference on Information Society (IS), Collaboration, Software And Services In Information Society (CSS), vol. A, 2013, pp. 211–214.

[26] Argouml. http://argouml.tigris.org/. Accessed: 2019-11-10.

[27] T. F. Bissyande´, F. Thung, S. Wang, D. Lo, L. Jiang, and L. Re´veille`re, "Empirical evaluation of bug linking," in 17th European Conference on Software Maintenance and Reengineering, CSMR 2013, Genova, Italy, March 5-8, 2013. IEEE Computer Society, 2013, pp. 89–98.

[28] Graphviz. http://www.graphviz.org/. Accessed: 2019-11-10.

[29] N. A. Aboud, G. Are´valo, J. Falleri, M. Huchard, C. Tibermacine, C. Urtado, and S. Vauttier, "Automated architectural component classification using concept lattices," in Joint Working IEEE/IFIP Conference on Software Architecture 2009 and European Conference on Software Architecture 2009, WICSA/ECSA 2009, Cambridge, UK, 14-17 September 2009. IEEE Computer Society, 2009, pp. 21–30.

[30] R. Wettel, M. Lanza, and R. Robbes, "Software systems as cities: a controlled experiment," in Proceedings of the 33rd International Conference on Software Engineering, ICSE 2011, Waikiki, Honolulu , HI, USA, May 21-28, 2011. ACM, 2011, pp. 551–560.

[31] M. Hammad and A. Rawashdeh, "A framework to measure and visualize class coupling," International Journal of Software Engineering and Its Applications, vol. 8, no. 4, pp. 137 – 146, 2014.

[32] M. Hammad and J. Cook, "Lightweight deployable software monitoring for sensor networks," in Proceedings of the 18th International Conference on Computer Communications and Networks, IEEE ICCCN 2009, San Francisco, California, USA, August 3-6, 2009. IEEE Computer Society, 2009, pp. 1–6.

[33] M. Hammad and J. Cook, "Lightweight monitoring of sensor software," in Proceedings of the 2009 ACM Symposium on Applied Computing (SAC), Honolulu, Hawaii, USA, March 9-12, 2009. ACM, 2009, pp. 2180–2185.

[34] M. Hammad, S. Abufakher, and M. Hammad, "A visualization approach for bug reports in software systems," International Journal of Software Engineering and Its Applications, vol. 8, no. 10, pp. 37 – 46, 2014.

[35] R. Al-Msie'deen, A. Seriai, M. Huchard, C. Urtado, and S. Vauttier, "Mining features from the object-oriented source code of software variants by combining lexical and structural similarity," in IEEE 14th International Conference on Information Reuse & Integration, IRI 2013, San Francisco, CA, USA, August 14-16, 2013. IEEE Computer Society, 2013, pp. 586–593.

[36] R. Al-Msie'deen, M. Huchard, A. Seriai, C. Urtado, and S. Vauttier, "Reverse engineering feature models from software configurations using formal concept analysis," in Proceedings of the Eleventh International Conference on Concept Lattices and Their Applications, Kos˘ice, Slovakia, October 7-10, 2014. CEUR-WS.org, 2014, pp. 95–106.

[37] R. Al-Msie'deen, A. Seriai, M. Huchard, C. Urtado, S. Vauttier, and H. E. Salman, "Mining features from the object- oriented source code of a collection of software variants using formal concept analysis and latent semantic indexing," in The 25th International Conference on Software Engineering and Knowledge Engineering, Boston, MA, USA, June 27-29, 2013. Knowledge Systems Institute Graduate School, 2013, pp. 244–249.

[38] R. Al-Msie'deen, A. Seriai, M. Huchard, C. Urtado, S. Vauttier, and H. E. Salman, "Feature location in a collection of software product variants using formal concept analysis," in Safe and Secure Software Reuse - 13th International Conference on Software Reuse, ICSR 2013, Pisa, Italy, June 18-20. Proceedings. Springer, 2013, pp. 302–307.

[39] R. Al-Msie'deen, "Reverse engineering feature models from software variants to build software product lines: REVPLINE approach," Ph.D. dissertation, Montpellier 2 University, France, 2014.

[40] R. Al-Msie'deen, A. Seriai, M. Huchard, C. Urtado, and S. Vauttier, "Documenting the mined feature implementations from the object-oriented source code of a collection of software product variants," in The 26th International Conference on Software Engineering and Knowledge Engineering, Hyatt Regency, Vancouver, BC, Canada, July 1-3, 2013. Knowledge Systems Institute Graduate School, 2014, pp. 138–143.

[41] R. Al-Msie'deen, M. Huchard, A. Seriai, C. Urtado, and S. Vauttier, "Automatic

documentation of [mined] feature implementations from source code elements and use-case diagrams with the REVPLINE approach," International Journal of Software Engineering and Knowledge Engineering, vol. 24, no. 10, pp. 1413–1438, 2014.

[42] M. Hammad, M. Hammad, and M. Bsoul, "An approach to automatically enforce object-oriented constraints," IJCAT, vol. 49, no. 1, pp. 50–59, 2014.

[43] H. Graham, H. Y. Yang, and R. Berrigan, "A solar system metaphor for 3D visualisation of object oriented software metrics," in Australasian Symposium on Information Visualisation, InVis.au, Christchurch, New Zealand, 23-24 January 2004, ser. CRPIT, vol. 35. Australian Computer Society, 2004, pp. 53–59.

[44] S. Haiduc, J. Aponte, L. Moreno, and A. Marcus, "On the use of automated text summarization techniques for summarizing source code," in 17th Working Conference on Reverse Engineering, WCRE 2010, 13-16 October 2010, Beverly, MA, USA. IEEE Computer Society, 2010, pp. 35–44.

[45] S. Haiduc, J. Aponte, and A. Marcus, "Supporting program comprehension with source code summarization," in Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 2, ICSE 2010, Cape Town, South Africa, 1-8 May 2010. ACM, 2010, pp. 223–226.

[46] R. Al-Msie'deen, "Automatic labeling of the object-oriented source code: The Lotus approach," Science International- Lahore, vol. 30, no. 1, pp. 45–48, 2018.

[47] A. D. Lucia, M. D. Penta, R. Oliveto, A. Panichella, and S. Panichella, "Labeling source code with information retrieval methods: an empirical study," Empirical Software Engineering, vol. 19, no. 5, pp. 1383–1420, 2014.

[48] S. C. B. de Souza, N. Anquetil, and K. M. de Oliveira, "A study of the documentation essential to software maintenance," in Proceedings of the 23rd Annual International Conference on Design of Communication: documenting & Designing for Pervasive Information, SIGDOC 2005, Coventry, UK, September 21-23, 2005. ACM, 2005, pp. 68–75.

[49] A. D. Lucia, M. D. Penta, R. Oliveto, A. Panichella, and S. Panichella, "Using IR methods for labeling source code artifacts: Is it worthwhile?" in IEEE 20th International Conference on Program Comprehension, ICPC 2012, Passau, Germany, June 11-13, 2012. IEEE Computer Society, 2012, pp. 193–202.

[50] B. Dit, L. Guerrouj, D. Poshyvanyk, and G. Antoniol, "Can better identifier splitting techniques help feature location?" in The 19th IEEE International Conference on

Program Comprehension, ICPC 2011, Kingston, ON, Canada, June 22-24, 2011. IEEE Computer Society, 2011, pp. 11–20.

[51] Wordnet. https://wordnet.princeton.edu/. Accessed: 2019-11-10.

[52] A. Kuhn, "Automatic labeling of software components and their evolution using log-likelihood ratio of word frequencies in source code," in Proceedings of the 6th International Working Conference on Mining Software Repositories, MSR 2009 (Co-located with ICSE), Vancouver, BC, Canada, May 16-17, 2009, Proceedings. IEEE Computer Society, 2009, pp. 175–178.

[53] R. Al-Msie'deen, M. Huchard, and C. Urtado, Reverse Engineering Feature Models. LAP LAMBERT Academic Publishing, 2014.

[54] A. Kuhn, S. Ducasse, and T. Gˆırba, "Semantic clustering: Identifying topics in source code," Information & Software Technology, vol. 49, no. 3, pp. 230–243, 2007.

[55] P. Warintarawej, M. Huchard, M. Lafourcade, A. Laurent, and P. Pompidor, "Software understanding: Automatic classification of software identifiers," Intell. Data Anal., vol. 19, no. 4, pp. 761–778, 2015.

[56] A. Wiese, V. Ho, and E. Hill, "A comparison of stemmers on source code identifiers for software search," in IEEE 27th International Conference on Software Maintenance, ICSM 2011, Williamsburg, VA, USA, September 25-30, 2011. IEEE Computer Society, 2011, pp. 496–499.

[57] M. V. Couto, M. T. Valente, and E. Figueiredo, "Extracting software product lines: A case study using conditional compilation," in 15th European Conference on Software Maintenance and Reengineering, CSMR 2011, 1-4 March 2011, Oldenburg, Germany. IEEE Computer Society, 2011, pp. 191–200.

[58] J. Emerson, N. Churcher, and A. Cockburn, "Tag clouds for software and information visualisation," in Prōceedings of the 14th Annual ACM SIGCHI NZ conference on Computer-Human Interaction, CHINZ 2013, Christchurch, New Zealand, November 15 - 16, 2013. ACM, 2013, pp. 1:1–1:4.

[59] R. Al-Msie'deen and A. Blasi, "The impact of the object-oriented software evolution on software metrics: The Iris approach," Indian Journal of Science and Technology, vol. 11, no. 8, pp. 1–8, 2018.

[60] N. Drouin, M. Badri, and F. Toure´, "Analyzing software quality evolution using metrics: An empirical study on open source software," JSW, vol. 8, no. 10, pp. 2462–2473, 2013.

[61] M. M. Lehman, "Laws of software evolution revisited," in Software Process Technology, 5th European Workshop, EWSPT '96, Nancy, France, October 9-11, 1996, Proceedings. Springer, 1996, pp. 108–124.

[62] K. Johari and A. Kaur, "Effect of software evolution on software metrics: an open source case study," ACM SIGSOFT Software Engineering Notes, vol. 36, no. 5, pp. 1–8, 2011.

[63] T. Kaur, N. Ratti, and P. Kaur, "Applicability of Lehman laws on open source evolution: A case study," International Journal of Computer Applications, vol. 93, no. 18, pp. 40–46, May 2014.

[64] S. R. Chidamber and C. F. Kemerer, "A metrics suite for object oriented design," IEEE Trans. Software Eng., vol. 20, no. 6, pp. 476–493, 1994.

[65] A. Israeli and D. G. Feitelson, "The Linux kernel as a case study in software evolution," Journal of Systems and Software, vol. 83, no. 3, pp. 485–501, 2010.

[66] J. Martinez, T. Ziadi, T. F. Bissyande´, J. Klein, and Y. L. Traon, "Automating the extraction of model-based software product lines from model variants," in 30th IEEE/ACM International Conference on Automated Software Engineering, ASE 2015, Lincoln, NE, USA, November 9-13, 2015. IEEE Computer Society, 2015, pp. 396–406.

[67] R. Al-Msie'deen, A.-D. Seriai, and M. Huchard, Reengineering Software Product Variants Into Software Product Line: REVPLINE Approach. LAP LAMBERT Academic Publishing, 2014.

[68] Rhino home page. https://developer.mozilla.org/en-US/docs/Mozilla/Projects/Rhino. Accessed: 2019-11-10.

[69] Mobile media home page. http://homepages.dcc.ufmg.br/~figueiredo/spl/icse08/. Accessed: 2019-11-10.

[70] E. Figueiredo, N. Cacho, C. Sant'Anna, M. Monteiro, U. Kulesza, A. Garcia, S. Soares, F. C. Ferrari, S. S. Khan, F. C. Filho, and F. Dantas, "Evolving software product lines with aspects: an empirical study on design stability," in 30th International Conference on Software Engineering (ICSE 2008), Leipzig, Germany, May 10-18, 2008. ACM, 2008, pp. 261–270.

[71] ArgoUML-SPL website. http://argouml-spl.tigris.org/. Accessed: 2019-11-10.

[72] R. Al-Msie'deen, M. Huchard, A. Seriai, C. Urtado, S. Vauttier, and A. Al-Khlifat, "Concept lattices: A representation space to structure software variability," in 2014 5th International Conference on Information and Communication Systems (ICICS). IEEE,

April 2014, pp. 1–6.

[73] R. Al-Msie'deen and A. Blasi, "Supporting software documentation with source code summarization," International Journal of Advanced and Applied Sciences, vol. 6, no. 1, pp. 59–67, 2019.

[74] T. Roehm, R. Tiarks, R. Koschke, and W. Maalej, "How do professional developers comprehend software?" in Software Engineering 2014, Fachtagung des GI-Fachbereichs Softwaretechnik, 25. February - 28. February 2014, Kiel, Deutschland. GI, 2014, p. 47.

[75] P. W. McBurney and C. McMillan, "Automatic source code summarization of context for java methods," IEEE Trans. Software Eng., vol. 42, no. 2, pp. 103–119, 2016.

[76] P. W. McBurney and C. McMillan, "An empirical study of the textual similarity between source code and source code summaries," Empirical Software Engineering, vol. 21, no. 1, pp. 17–42, 2016.

[77] S. S. Yau and J. S. Collofello, "Some stability measures for software maintenance," IEEE Trans. Software Eng., vol. 6, no. 6, pp. 545–552, 1980.

[78] N. Dave, D. B. Davis, K. Potts, and H. U. Asuncion, "Uncovering file relationships using association mining and topic modeling," in The Sixth International Conference on Information, Process, and Knowledge Management, 2014, pp. 105– 111.

[79] L. Moreno, J. Aponte, G. Sridhara, A. Marcus, L. L. Pollock, and K. Vijay-Shanker, "Automatic generation of natural language summaries for java classes," in IEEE 21st International Conference on Program Comprehension, ICPC 2013, San Francisco, CA, USA, 20-21 May, 2013. IEEE Computer Society, 2013, pp. 23–32.

[80] L. Shi, H. Zhong, T. Xie, and M. Li, "An empirical study on evolution of API documentation," in Fundamental Approaches to Software Engineering - 14th International Conference, FASE 2011, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2011, Saarbru¨cken, Germany, March 26-April 3, 2011. Proceedings. Springer, 2011, pp. 416–431.

[81] A. J. Ko, B. A. Myers, M. J. Coblenz, and H. H. Aung, "An exploratory study of how developers seek, relate, and collect relevant information during software maintenance tasks," IEEE Trans. Software Eng., vol. 32, no. 12, pp. 971–987, 2006.

[82] T. D. LaToza, G. Venolia, and R. DeLine, "Maintaining mental models: a study of developer work habits," in 28th International Conference on Software Engineering (ICSE 2006), Shanghai, China, May 20-28, 2006. ACM, 2006, pp. 492–501.

[83] D. R. Radev, E. H. Hovy, and K. R. McKeown, "Introduction to the special issue on summarization," Computational Linguistics, vol. 28, no. 4, pp. 399–408, 2002.

[84] G. Sridhara, E. Hill, D. Muppaneni, L. L. Pollock, and K. Vijay-Shanker, "Towards automatically generating summary comments for java methods," in ASE 2010, 25th IEEE/ACM International Conference on Automated Software Engineering, Antwerp, Belgium, September 20-24, 2010. ACM, 2010, pp. 43–52.

[85] L. Moreno, A. Marcus, L. L. Pollock, and K. Vijay-Shanker, "Jsummarizer: An automatic generator of natural language summaries for java classes," in IEEE 21st International Conference on Program Comprehension, ICPC 2013, San Francisco, CA, USA, 20-21 May, 2013. IEEE Computer Society, 2013, pp. 230–232.

[86] L. Moreno and A. Marcus, "Automatic software summarization: the state of the art," in Proceedings of the 40th International Conference on Software Engineering: Companion Proceedings, ICSE 2018. ACM, 2018, pp. 530–531.

[87] Y. Kanellopoulos, T. Dimopulos, C. Tjortjis, and C. Makris, "Mining source code elements for comprehending object- oriented systems and evaluating their maintainability," SIGKDD Explorations, vol. 8, no. 1, pp. 33–40, 2006.

[88] Nanoxml. http://nanoxml.sourceforge.net/orig/index.html. Accessed: 2019-11-10.

[89] Argouml. http://argouml-downloads.tigris.org/argouml-0.28.1/. Accessed: 2019-11-10.

[90] Nanoxml JavaDocs. http://nanoxml.sourceforge.net/orig/NanoXML-2-JavaDoc/. Accessed: 2019-11-10.

[91] Argouml JavaDocs. http://argouml-stats.tigris.org/nonav/javadocs/javadocs-0.28/. Accessed: 2019-11-10.

[92] R. Al-Msie'deen, "Tag clouds for object-oriented source code visualization," Engineering, Technology & Applied Science Research, vol. 9, no. 3, pp. 4243–4248, 2019.

[93] O. Kaser and D. Lemire, "Tag-cloud drawing: Algorithms for cloud visualization," In: Proceedings of Tagging and Metadata for Social Information Organization (WWW 2007), https://arxiv.org/abs/cs/0703109. 2007.

[94] M. A. Hearst and D. K. Rosner, "Tag clouds: Data analysis tool or social signaller?" in 41st Hawaii International Conference on Systems Science (HICSS-41 2008), Proceedings, 7-10 January 2008, Waikoloa, Big Island, HI, USA. IEEE Computer Society, 2008, p. 160.

[95] M. J. Wilson and M. L. Wilson, "Tag clouds and keyword clouds: evaluating zero-interaction benefits," in Proceedings of the International Conference on Human Factors in Computing Systems, CHI 2011, Extended Abstracts Volume, Vancouver, BC, Canada, May 7-12, 2011. ACM, 2011, pp. 2383–2388.

[96] W. Cui, Y. Wu, S. Liu, F. Wei, M. X. Zhou, and H. Qu, "Context-preserving, dynamic word cloud visualization," IEEE Computer Graphics and Applications, vol. 30, no. 6, pp. 42–53, 2010.

[97] L. J. García-Castro, M. Hepp, and A. G. Castro, "Tags4tags: Using tagging to consolidate tags," in Database and Expert Systems Applications, 20th International Conference, DEXA 2009, Linz, Austria, August 31 - September 4, 2009. Proceedings. Springer, 2009, pp. 619–628.

[98] G. J. Greene and B. Fischer, "Interactive tag cloud visualization of software version control repositories," in 3rd IEEE Working Conference on Software Visualization, VISSOFT 2015, Bremen, Germany, September 27-28, 2015. IEEE Computer Society, 2015, pp. 56–65.

[99] J. Emerson, N. Churcher, and C. Deaker, "From toy to tool: Extending tag clouds for software and information visualisation," in 22nd Australian Conference on Software Engineering (ASWEC 2013), 4-7 June 2013, Melbourne, Victoria, Australia. IEEE Computer Society, 2013, pp. 155–164.

[100] C. Anslow, J. Noble, S. Marshall, and E. D. Tempero, "Visualizing the word structure of java class names," in Companion to the 23rd Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, OOPSLA 2008, October 19-13, 2007, Nashville, TN, USA. ACM, 2008, pp. 777–778.

[101] R. Cottrell, B. Goyette, R. Holmes, R. J. Walker, and J. Denzinger, "Compare and contrast: Visual exploration of source code examples," in Proceedings of the 5th IEEE International Workshop on Visualizing Software for Understanding and Analysis, VISSOFT 2009, Edmonton, Alberta, Canada, September 25, 2009. IEEE Computer Society, 2009, pp. 29–32.

[102] I. Ada¨, K. Thiel, and M. R. Berthold, "Distance aware tag clouds," in Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Istanbul, Turkey, 10-13 October 2010. IEEE, 2010, pp. 2316–2322.

[103] J. Martinez, T. Ziadi, T. F. Bissyande´, J. Klein, and Y. L. Traon, "Name suggestions during feature identification: the variclouds approach," in Proceedings of the 20th

International Systems and Software Product Line Conference, SPLC 2016, Beijing, China, September 16-23, 2016. ACM, 2016, pp. 119–123.

[104] Sourcecloud. https://misto.ch/2011/09/19/tag-cloud-visualization-for-source-code/. Accessed: 2019-11-10.

[105] Wordle. http://www.wordle.net/. Accessed: 2019-11-10.

[106] G. A. Miller, "Wordnet: A lexical database for english," Communications of the ACM, vol. 38, no. 11, pp. 39–41, 1995.

[107] J. Emerson, "Tag clouds in software visualisation," University of Canterbury, 2014.

[108] J. Martinez, N. Ordonez, X. Ternava, T. Ziadi, J. Aponte, E. Figueiredo, and M. T. Valente, "Feature location benchmark with ArgoUML SPL," in Proceedings of the 22nd International Systems and Software Product Line Conference - Volume 1, SPLC 2018, Gothenburg, Sweden, September 10-14, 2018. ACM, 2018, pp. 257–263.

[109] M. Halvey and M. T. Keane, "An assessment of tag presentation techniques," in Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007. ACM, 2007, pp. 1313–1314.

[110] S. Lohmann, J. Ziegler, and L. Tetzlaff, "Comparison of tag cloud layouts: Task-related performance and visual exploration," in Human-Computer Interaction - INTERACT 2009, 12th IFIP TC 13 International Conference, Uppsala, Sweden, August 24-28, 2009, Proceedings, Part I. Springer, 2009, pp. 392–404.