**GE3 Bibliography**

**G89 T08**

**Eric Feng (100531465)**

AlOmar, Eman Abdullah, et al. “Automating Source Code Refactoring in the Classroom.”

Proceedings of the 55th ACM Technical Symposium on Computer Science Education V.

1, ACM, 7 Mar. 2024. Crossref, doi:10.1145/3626252.3630787.

This paper explores the integration of automated refactoring tools into software engineering education. It focuses on how using the JDeodorant IDE plugin, which assists in detecting and correcting antipatterns, can enhance students' understanding of software quality and refactoring practices. The study is relevant to software refactoring as it demonstrates how automated tools can be effectively used to teach refactoring concepts with real-world practice. By conducting experiments within the classroom, the researchers show that students not only improve their ability to identify and correct poor coding practices but also appreciate the tools, enabling a deeper understanding of software maintenance and improvement. This is useful in educational settings to prepare students for real-world software development challenges.

Almogahed, Abdullah, et al. “A Refactoring Classification Framework for Efficient Software

Maintenance.” IEEE Access, vol. 11, Institute of Electrical and Electronics Engineers

(IEEE), 2023, pp. 78904–17. Crossref, doi:10.1109/access.2023.3298678.

This paper investigates the financial cost associated with software maintenance and proposes a framework to optimize refactoring practices. The authors address the inconsistencies in how different refactoring techniques impact software quality attributes. By proposing a classification framework that categorizes refactoring techniques based on their measurable effects on internal quality attributes, the study aims to assist developers in making informed decisions about which techniques to use to meet specific design objectives. This framework helps in enhancing software quality and efficiency in maintenance efforts for software engineers.

]

**Kevin Liao (100531529)**

Abid, C., Alizadeh, V., Kessentini, M., Ferreira, T. D., & Dig, D. (2020). 30 Years of Software Refactoring Research: A Systematic Literature Review. *ArXiv*. /abs/2007.02194

This paper provides a systematic literature review of the past 30 years of software refactoring research. Software development has evolved drastically during this time, leading to refactoring studies expanding beyond traditional code-level modifications. Today a diverse set of levels, objectives, and domains leverage refactoring, leading to refactoring research efforts needing to be more cohesive across the field. The researchers address this by introducing a taxonomy to categorize refactoring research, highlighting different trends and gaps in the literature. This guides the direction of future research and emphasizes the broad application and importance of refactoring techniques in enhancing all software.

Shirafuji, A., Oda, Y., Suzuki, J., Morishita, M., & Watanobe, Y. (2023). Refactoring Programs

Using Large Language Models with Few-Shot Examples. *ArXiv*.

https://doi.org/10.1109/APSEC60848.2023.00025

This paper proposes using large language models (LLM) to assist software developers in refactoring software to reduce complexity and increase maintainability. By providing the model with few-shot examples, LLMs are proven to simplify code effectively without altering the program’s functionality. The introduction of AI in refactoring programs will allow engineers to enhance readability and maintainability in their code while adhering to traditional coding standards. The researchers provide qualitative and quantitative data highlighting the beneficial impact LLMs have on code syntax, structure, and formatting. This marks a pivotal shift as AI is combined with traditional software development practices to optimize program refactoring.

**José Estébanez de la Torre (100495697)**

Citation: Ministry of Testing. (n.d.). Ministry of Testing. Retrieved from https://www.ministryoftesting.com/

Ministry of Testing is a prominent online platform dedicated to software testing professionals. It offers a wealth of resources including articles, tutorials, courses, and forums covering a wide range of testing topics. With a vibrant community of testers, Ministry of Testing provides valuable insights, best practices, and tools to enhance testing skills and knowledge.

Citation: The Testing Planet. (n.d.). The Testing Planet. Retrieved from https://www.testingcircus.com/category/the-testing-planet/

The Testing Planet is a digital magazine focusing on software testing. It features articles, insights, and opinions from software testing professionals worldwide. Covering industry trends, emerging technologies, and best practices, The Testing Planet provides valuable resources for testers looking to stay updated and informed in the dynamic field of software testing.