



Universidad
Carlos III de Madrid

EG3

AUTORES:

Iván Ciller López: 100522245
100522245@alumnos.uc3m.es

Sergio Villafuertes Jiménez: 100522323
100522323@alumnos.uc3m.es

GRUPO: 84

Desarrollo de software

Índice

Nombres	4
REGLA N°1	4
REGLA N°2	4
REGLA N°3	4
REGLA N°4	4
REGLA N°5	5
REGLA N°6	5
Atributos	5
REGLA N°7	5
REGLA N°8	5
Variables	6
REGLA N°9	6
Métodos	6
REGLA N°10	6
Línea de código	6
REGLA N°11	6

Artículo 1 (100522323)

Referencia

Título: *How to Refactor this Code? An Exploratory Study on Developer-ChatGPT Refactoring Conversations*

Publicación: 2024 IEEE/ACM 21st International Conference on Mining Software Repositories (MSR)

Web: <https://dl.acm.org/doi/pdf/10.1145/3643991.3645081>

Autores: Eman Abdullah AlOmar, Anushkrishna Venkatakrishnan, Mohamed Wiem Mkaouer, Christian D. Newman y Ali Ouni.

Fecha de publicación: 02/07/2024

Resumen

Este artículo es relevante ya que profundiza en la interacción entre desarrolladores y ChatGPT en el proceso de refactorización de código, analizando más de 17,000 conversaciones reales. A través de este estudio, se identifican patrones en la manera en que los desarrolladores solicitan mejoras, así como los criterios de calidad que el modelo prioriza, tales como legibilidad, mantenibilidad y simplicidad. Los hallazgos resaltan tanto el potencial como las limitaciones de los modelos de lenguaje en la optimización de código, proporcionando valiosas claves para su aplicación más efectiva en el desarrollo de software.

Artículo 2 (100522323)

Referencia

Título: *Behind the Intent of Extract Method Refactoring: A Systematic Literature Review*

Publicación: IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 50, NO. 4, APRIL 2024

Web: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10381511>

Autores: Eman Abdullah AlOmar, Mohamed Wiem Mkaouer y Ali Ouni.

Fecha de publicación: 04/01/2024

Resumen

Este artículo es relevante ya que presenta una revisión detallada sobre la técnica de refactorización Extract Method, considerada una de las más utilizadas en ingeniería de software. A partir del análisis de 83 estudios primarios, los autores exploran las razones detrás de su aplicación, los enfoques técnicos empleados, el nivel de automatización y las herramientas desarrolladas para facilitar su implementación, así como otro tipo de características. Además, el estudio identifica desafíos y oportunidades de mejora en el diseño de herramientas de recomendación, ofreciendo una referencia valiosa tanto para la investigación futura como para su aplicación práctica en el desarrollo de software.

Artículo 3 (100522245)

Referencia

Título: *Exploring the potential of general purpose LLMs in automated software refactoring: an empirical study*

Publicación: Automated Software Engineering, Volumen 32, Artículo número 26 (2025)

Web: <https://link.springer.com/article/10.1007/s10515-025-00500-0>

Autores: Bo Liu, Yanjie Jiang, Yuxia Zhang, Nan Niu, Guangjie Li, Hui Liu

Fecha de publicación: 01/03/2025

Resumen

Este artículo investiga el potencial de los modelos de lenguaje de propósito general (LLMs), en concreto ChatGPT y Gemini, en la automatización del refactorizado de software. Para ello, se crearon un conjunto de datos de alta calidad con 180 refactorizaciones reales de 20 proyectos. Los resultados indican que, sin orientación específica, los LLMs tienen un rendimiento limitado en la identificación de oportunidades de refactorización. Sin embargo, al proporcionar subcategorías de refactorización y reducir el espacio de búsqueda en las indicaciones, la tasa de éxito de ChatGPT aumentó significativamente del 15.6% al 86.7%. En cuanto a la recomendación de soluciones de refactorización, ChatGPT propuso soluciones para 176 de las 180 refactorizaciones, siendo el 63.6% comparables o superiores a las realizadas por expertos humanos. No obstante, algunas soluciones generadas por los LLMs presentaron riesgos, como cambios en la funcionalidad del código o errores de sintaxis, lo que remarca la necesidad de precaución en su aplicación.

Artículo 4 (100522245)

Referencia

Título: *A Novel Refactoring and Semantic Aware Abstract Syntax Tree Differencing Tool and a Benchmark for Evaluating the Accuracy of Diff Tools*

Publicación: ACM Transactions on Software Engineering and Methodology (TOSEM), Volumen 34, Número 2

Web: <https://dl.acm.org/doi/10.1145/3696002>

Autores: Pouria Alikhanifard, Nikolaos Tsantalis

Fecha de publicación: 24/01/2025

Resumen

Este artículo presenta una nueva herramienta de diferenciación de árboles de sintaxis abstracta (AST) que incorpora conciencia semántica y de refactorización. Además, se introduce un conjunto de datos de referencia para evaluar la precisión de las herramientas de diferenciación de código. La herramienta propuesta mejora la detección de cambios

semánticos y refactorizaciones en el código fuente, superando las limitaciones de las herramientas existentes. Los autores también proporcionan una evaluación exhaustiva utilizando el conjunto de datos de referencia, demostrando la eficacia y precisión de su enfoque en comparación con las soluciones actuales.