

Information system for automating the process of checking student works requiring software code development					
<b>Article:</b>	<i>Maertens, Rien, Peter Dawyndt, and Bart Mesuere. "Dolos 2.0: Towards Seamless Source Code Plagiarism Detection in Online Learning Environments." Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 2. 2023.</i>	<i>Pankiewicz, Maciej, and Ryan S. Baker. "Large Language Models (GPT) for automating feedback on programming assignments." arXiv preprint arXiv: 2307.00150 (2023).</i>	<i>Ludwig, Jeremy, and Devin Cline. "Challenges in Explaining Source Code Quality Assessment." 2022 IEEE Aerospace Conference (AERO). IEEE, 2022.</i>	<i>Fabijanić, M., et al. "Automatic Evaluation of Student Software Solutions in a Virtualized Environment." 2023 46th MIPRO ICT and Electronics Convention (MIPRO). IEEE, 2023.</i>	<i>Khanzadeh, Sourena, et al. "Opti Code Pro: A Heuristic Search-based Approach to Code Refactoring." arXiv preprint arXiv:2305.07594 (2023).</i>
<b>Link:</b>	<a href="https://dl.acm.org/doi/abs/10.1145/3587103.3594166">https://dl.acm.org/doi/abs/10.1145/3587103.3594166</a>	<a href="https://arxiv.org/abs/2307.00150">https://arxiv.org/abs/2307.00150</a>	<a href="https://ieeexplore.ieee.org/abstract/document/9843840/">https://ieeexplore.ieee.org/abstract/document/9843840/</a>	<a href="https://ieeexplore.ieee.org/abstract/document/10159927/">https://ieeexplore.ieee.org/abstract/document/10159927/</a>	<a href="https://arxiv.org/abs/2305.07594">https://arxiv.org/abs/2305.07594</a>
<b>Themes:</b>					
<b>Methods of combating code plagiarism</b>	Development of a new source code plagiarism detection tool named Dolos, which supports a wide range of programming languages and is designed to be user-friendly.			The proposed assessment model analyzes key parts of the student solution source codes and commands against a combination of test case input data using the pattern-finding method.	
<b>Code quality assessment</b>			The focus is on discussing how CBR-Insight communicates source code quality assessment to the development team, including non-technical decision makers and software developers.	Generic security models such as user-level restrictions, process-level restrictions, and virtualization have been proposed to handle code executions in a secure way.	
<b>Automated feedback generation for programming assignments</b>		The paper focuses on automated feedback generation using OpenAI's GPT-3.5 model for programming assignments. The study shows that students rated the usefulness of GPT-generated hints positively and the experimental group, which had GPT hints enabled, performed better in terms of successful submissions. However, there was a potential over-reliance on GPT-generated feedback when it was unavailable.	Analyzed over 45,000 student submissions to programming exercises and used an automated assessment tool to provide feedback based on test results. They monitored the progress that students made over the course of a session.		
<b>Student work evaluation software</b>	Dolos, the source code plagiarism detection tool, can be used by teachers to evaluate and assess student work in programming courses.			Automated assessment of students software decisions in a virtualised environment, particularly in the Programming course and relational database courses in College.	
<b>Code refactoring</b>		The availability of GPT hints positively impacted student performance, as the experimental group (with GPT hints enabled) performed better in terms of the percentage of successful submissions across consecutive attempts for tasks where GPT hints were enabled.			Approach to code refactoring using best-first search methods. It proposes using heuristic search-based techniques to guide the refactoring process towards solutions with high cohesion and low coupling. The approach was evaluated using demonstrative examples and a tool was created to implement the algorithm on Java projects.