Software modeling notations such as the Unified Modeling Language and the Business Process Model and Notation occupy a key role in software engineering education; however, this importance is not reflected in professional practice, where such notations are underused, and students perceive learning them as a daunting and frustrating task. The reasons behind this gap include the cognitive effort needed to correctly map informal requirements to formal graphical representation, and the necessity of immediate, personalized, and formative feedback to foster development; a necessity that is, however, hard to cover in large-scale university courses.

This PhD thesis investigates whether two approaches can be applied to modeling education to address these limitations. The first, methodological approach is the use of gamification, the act of integrating game design elements in non-recreational contexts, to guide the development of modeling tools that integrate automated diagram assessment with motivational mechanics, while the second approach is technological and consists of the use of large language models as intelligent agents that can generate modeling artifacts, to support the evaluation systems of gamified tools and provide educational assistance.

The gamification aspects of the research are grounded in four empirical studies performed with two tools developed throughout the first years of the PhD. One of them, named BIPMIN and focused on gamifying BPMN modeling, was first assessed with a preliminary usability evaluation and then used in a controlled experiment with students of a Masters level course. The findings from the experiment led to the definition of a generalized gamification framework that then guided the development of UMLegend, a tool that supports UML class diagram assignments, which was then evaluated with a second controlled experiment, following the same design as the first one. The BIPMIN tool was then redesigned according to the framework and adopted for a longitudinal study through the 2023-2024 edition of an Information Systems course, tracking the impact of the game mechanics on modeling performance over five progressive exercises and through grade comparison with the preceding non-gamified cohort.

The most consistent finding across all four studies is the positive effect of mechanics that provide direct element-level guidance such as completeness indicators, in-diagram feedback that changes the style of correct/wrong elements, and detailed error lists, with these elements improving diagram correctness, being more appreciated by students, and making students more willing to interact with the evaluation loop, regardless of notation or duration of the interaction. Furthermore, the longitudinal study proved that gamification leads to significant improvements in the final course evaluations, while also bringing a progressive reduction of errors through the course, demonstrating that gamified modeling education can bring benefits even over longer learning periods. However, long-term engagement mechanics proved to be less appreciated in the longitudinal setting as well as in the single-session experiments, hinting that such mechanics may require additional social implementations to provide motivational effects.

The research activities focused on LLMs instead covered three empirical studies and the development of a prototype tool between 2023 and 2025. Two comparative studies evaluated the GPT-4 architecture against a trained human solver on, respectively, UML class diagram and use case diagram exercises; then, an LLM-assisted tool named TIGRE was developed, intended to build on the findings of the two comparative studies and to provide

teacher support in the definition of reference solutions for UML class diagram exercises, with the end goal of combining it with the aforementioned UMLegend. Lastly, a final comparative study continued the research thread on class diagram generation by testing the performance of four LLM agents, with a mix of proprietary and open source models and adopting a role-based few-shot prompting strategy.

Findings across the three studies on LLM-assisted modeling tasks show that general-purpose models can be compared to students in terms of syntactic quality, while performing significantly worse in terms of representing semantic, domain-dependent requirements on class diagrams. Performance proved to be more comparable on the structurally simpler use case notation. The use of advanced prompting techniques can remove syntax errors entirely for most models, while keeping the semantic weakness, an issue that is encountered in all four architectures. Finally, the prototype assistant demonstrates that LLM-generated diagrams can be used as structural starting templates and reduce the effort needed to create reference solutions, while at the same time needing teacher intervention to address the inevitable semantic violations.

The thesis's contribution to the research literature on software modeling and model-based engineering consists of two developed and evaluated software artifacts (UMLegend, BIPMIN) focused on gamified modeling education, the TIGRE prototype for LLM-assisted diagram creation (still currently under development at the time of writing), a three-step automated evaluation pipeline that ties syntactic, semantic, and pragmatic assessment to gamified feedback, and a multi-notation multi-model empirical benchmark for LLM-generated diagrams. Together, the findings establish that gamification can reliably improve modeling education if it ties game mechanics to high-quality automated formative feedback, and that LLMs are effective drafting helpers for both diagram generation and exercise creation tasks, with semantic domain inference being a shared limitation for both research areas.