

Domain Diversity, Motivation, Inclusion, and Feedback in Software Modelling Education

Isabella Graßl
isabella.grassl@tu-darmstadt.de
TU Darmstadt
Darmstadt, Germany

Christopher Lazik
lazikchr@hu-berlin.de
HU Berlin
Berlin, Germany

Shalini Chakraborty
s.chakraborty@uni-bayreuth.de
Universität Bayreuth
Bayreuth, Germany

Grischa Liebel
grischal@ru.is
Reykjavik University
Reykjavik, Iceland

Miguel Goulão
mgoul@fct.unl.pt
NOVA-LINCS, NOVA
School of Science and
Technology
Lisbon, Portugal

Abstract

Student engagement is critical for effective learning in software modelling, yet fostering motivation and inclusivity remains a challenge. While existing research has focused on modelling tools, notations, and assessment, little attention has been given to how the choice of problem domains and the diversity, relatability, and cultural perspectives they bring shape students' learning experiences. This study explores how problem domains and teaching methods influence motivation, engagement, inclusiveness, and feedback in modelling education. To investigate these dimensions, we conducted parallel surveys with 90 students and 22 educators. Our findings reveal disconnects between educator assumptions and student preferences: Students show greatest motivation for socially relevant domains and prefer choice in selection, while educators overestimate interest in study-related domains. The study identifies how minor design choices can exclude students. Students perceive feedback as meaningful when visibly acted upon. These findings suggest inclusive domain selection is central to student motivation; thus, we recommend student-centred domain selection.

CCS Concepts

• **Software and its engineering** → **Software system models**; • **Social and professional topics** → **Computing education**;

Keywords

Diversity, Modelling, Software Engineering Education.

ACM Reference Format:

Isabella Graßl, Christopher Lazik, Shalini Chakraborty, Grischa Liebel, and Miguel Goulão. 2026. Domain Diversity, Motivation, Inclusion, and Feedback in Software Modelling Education. In *34th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (FSE Companion '26)*, July 05–09, 2026, Montreal, QC, Canada. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3803437.3805791>

1 Introduction

Software modelling is a core skill in software engineering (SE) education, enabling students to abstract, conceptualise, design, and manage complex systems [20, 30]. Despite its importance, software

modelling has often been under-emphasised in curricula [19]. Students may find modelling conceptually demanding [13], struggle with tools [1, 39] and perceive it as detached from practical work [5]. Educators, meanwhile, face limited class time and resources, and modelling is often taught in a top-down and formal way [58], making student engagement a challenge.

One promising way to motivate students in modelling education is the choice of example domains in assignments and lectures, as Paige et al. [45] observed: “*students both need and benefit from examples when learning how to model. Examples reinforce the conceptual principles and engineering practice of modelling*”. The choice of domains is not merely illustrative since relatable and interesting domains might help students connect abstract concepts to what they value. In contrast, traditional examples such as games or library management systems may not resonate with all students and can reduce engagement, contribute to exclusion, and harm learning outcomes [50]. However, problem domains are only one part of the learning experience. The way modelling is taught shapes how students engage with it. Teaching methods such as collaborative [42] or gamified [9, 24] learning can make modelling more meaningful.

Recent work highlights the importance of human factors in software modelling [40] and inclusion in modelling education [6], aiming to create classrooms where all students feel welcomed and represented. However, while there is substantial research on software modelling education on tools [1, 2, 39], student learning [3, 4, 41], assessment practices [15, 27], and classroom practices [23, 34], there is, to our knowledge, no work that examines how the choice of problem domains influences students. In this study, we investigate how problem domains and teaching methods influence motivation, engagement, inclusiveness, and feedback in modelling education. We address the following research questions (RQs):

RQ1: *How do different modelling domains affect student motivation and engagement?*

RQ2: *How do preferences for teaching methods and assignment design choices affect motivation in modelling assignments?*

RQ3: *What are the perceptions of inclusiveness in modelling assignments?*

RQ4: *What are the perceptions of feedback in modelling assignments?*

To answer the RQs, we conducted a survey with 22 educators and 90 students with modelling experience. Our findings show disagreement among students about which problem domains are motivating. While educators hold strong assumptions about which



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

FSE Companion '26, Montreal, QC, Canada

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-2636-1/2026/07

<https://doi.org/10.1145/3803437.3805791>

domains motivate students, students show much more diverse preferences driven primarily by personal interests and social relevance. This is the first study to compare educators' and students' perspectives on motivation, engagement, inclusiveness, and feedback in modelling assignments, providing insights that can help educators design more engaging and inclusive modelling courses.

2 Background and Related Work

We first examine challenges in teaching modelling, then explore what motivates students in this context, and finally consider how inclusion in teaching can address motivational barriers.

2.1 Software Modelling Education

While modelling is part of many SE curricula, pedagogical emphasis often remains superficial, focused on syntax rather than conceptual modelling and model-driven thinking. Whittle et al. [59] demonstrate the importance of hands-on modelling exercises, while critiquing the predominant focus on syntactic correction over real-world problem-solving. Educators in the MODELS community have repeatedly emphasised the lack of comprehensive teaching resources and well-established pedagogical approaches for modelling and model-driven engineering (MDE) education [18]. Gogolla and Stevens [25] conceptualise modelling education as an ecosystem, identifying the factors that constitute this ecosystem and describing their relationships. They highlight elements such as teaching domains and instructional styles as particularly influential in shaping effective modelling education. Modelling tools commonly used in education introduce *accidental complexity*: setup, configuration issues, and poor usability for classroom settings, distract students from understanding core modelling concepts [1, 39]. Recent guidelines emphasise the need for teaching-specific modelling tools with easy setup, low friction, and meaningful feedback [35]. The field's reliance on the Unified Modelling Language UML as the dominant modelling language is well-established in Seidl et al.'s [51] instructional methodology. However, Petre [48] provide empirical evidence that this academic focus does not match industrial practice, where UML adoption is often partial or tool-driven. This industry-academia gap is further quantified by Ciccozzi et al. [17], whose analysis of executable UML models reveals significant limitations in translating educational models to production systems. Empirical studies document students' struggles with fundamental modelling concepts such as abstraction, multiplicity, and mapping textual requirements to diagrams, highlighting prevalent misconceptions that hinder learning unless actively addressed [16]. Collectively, these challenges create substantial technological and cognitive barriers for students, which in turn diminish their motivation. This paper suggests concrete mitigation strategies for these challenges.

2.2 Student Motivation in Software Modelling

Motivation is critical in engineering education and can be framed through, e.g., Self-Determination Theory [54], which posits that autonomy, competence, and relatedness enhance motivation and learning. In modelling education, motivational barriers are often tied to tooling issues, delayed or unhelpful feedback or unclear expectations from the modelling task itself [2, 13, 35]. Prior work shows that lack of experience in the subject matter limits students'

ability to engage flexibly in modelling, and that when the domains are unfamiliar, this reduces confidence and contributes to frustration [12]. Some educators have turned to gamification, using game elements like badges or challenges, to boost engagement in modelling courses [10], though results vary depending on implementation. However, there is limited research examining the influence of problem domains in software modelling education, and particularly a lack of diversity in the example domains used. The results in this paper help bridge this gap.

2.3 Diversity in Problem Domains and Inclusion

Diversity in problem domains means exposing students to a variety of contexts and applications in modelling assignments, while inclusion ensures that these domains are relatable and do not privilege specific backgrounds or prior knowledge, contributing to culturally responsive computing [14, 50]. Inclusive pedagogies such as Universal Design for Instruction (UDI), culturally relevant teaching, and equitable assignment design advocate for multiple means of representation, flexible assessment, and structural responsiveness to diverse learner needs [11, 22, 37]. Active learning methods, such as model construction through guided, incremental tasks, have shown promise in improving engagement and conceptual grasp in UML contexts [55]. Pérez and Rubio [48] demonstrate that project-based learning (PBL) in UML education leads to higher scores on UML examinations compared to traditional learners, suggesting active engagement in modelling projects improves knowledge retention. PBL also enables the development of transferable skills, including teamwork and communication, suggesting it can increase student motivation without compromising performance in other course components. This aligns with Szmurło and Śmiałek's [52] project-based framework, which shows iterative modelling tasks improve conceptual understanding compared to lecture-based approaches.

Prior work highlights persistent challenges in teaching software modelling, which range from limited pedagogical resources and motivational barriers to insufficiently diverse and inclusive assignment domains, yet little is known about how these factors intersect to shape student engagement and learning. This paper offers insights into the role of these factors in teaching software modelling.

3 Method

We investigate how the choice of problem domains and teaching methods influence motivation, engagement, inclusiveness, and feedback in modelling education and how these vary across diverse student groups. We compare students' and educators' perceptions, to identify potential misconceptions or biases in teaching materials.

3.1 Instrumentation

We conducted two surveys: one for students and one for educators. Both surveys were designed to capture the current state of problem domains in software modelling assignments and how these domains influence motivation, engagement, inclusiveness, and feedback in modelling education. We relied on self-reported data to capture subjective perceptions, as lived experience and personal voice were central. Although self-reports may not fully align with objective outcomes, they are often closely linked to learning behaviours and achievements [28, 46].

Table 1: Survey questions for students and educators. (SC/MC = single/multiple choice, FT = free text, LI = Likert)

Var.	Question	Type	Target
Demographics			
DE01	What age are you?	SC	Both
DE02	What gender do you identify as?	SC	Both
DE04	In which country are you currently studying/teaching?	FT	Both
DE06	What degree are you currently pursuing?	SC	Students
DE07	How many years of work experience in software engineering do you have?	SC	Students
DE08	How many years of teaching experience in software engineering do you have?	SC	Educators
DE09	Which of the following minorities do you belong to?	MC	Both
DE10	Which courses/topics do you typically teach?	FT	Educators
DE11	Are you an active member in the modeling research community?	SC	Educators
DE12	How many years of experience in software modelling do you have?	SC	Students
Motivation → RQ1			
MO01	How motivated are you/do you think students are to work on projects related to the following domains?	–	Both
MO01-01	Video Games	5-pt LI	Both
MO01-02	Community Platforms (e.g., sharing applications, NGO platforms, social media)	5-pt LI	Both
MO01-03	Public Information Systems (e.g., traffic routing, flight booking)	5-pt LI	Both
MO01-04	Enterprise & Business Systems (e.g., financial reporting, human resource management)	5-pt LI	Both
MO01-05	Automation Systems (e.g., smart buildings, cyber-physical systems)	5-pt LI	Both
MO01-06	Others	5-pt LI	Both
MO02	Please briefly explain your choice in the previous question.	FT	Both
Design of Assignments → RQ2			
DA01	When designing modelling assignments, how do you select the domain?	FT	Educators
DA02	What domains do you typically use in assignments and/or lectures?	FT	Educators
DA04	How do you consider student interest or motivation when designing modelling assignments?	FT	Educators
Preferences for Teaching Methods → RQ2			
PR02	To what extent do you think gamification in education motivates students?	5-pt LI	Both
PR03	Please briefly explain your choice in the previous question.	FT	Both
PR04	Which kind of collaboration do you prefer in modelling assignments?	SC	Both
PR05	Please briefly explain your choice in the previous question.	FT	Both
PR06	How do you encourage students to engage with your modelling assignments?	FT	Educators
Inclusion and Bias → RQ3			
IN02	Have you ever felt that an assignment topic was excluding a specific sub-group of students?	FT	Students
IN03	Do you think the choice of problem domains in modelling assignments affects the inclusiveness of the course?	FT	Students
IN04	How do you feel your learning success is impacted by the choice of domain in modelling assignments?	FT	Students
IN01	Please provide examples of problem domains you believe are interesting for a diverse group of students.	FT	Students
EI01	How do you consider student diversity when selecting domains for modelling assignments?	FT	Educators
EI02	Have you observed any differences in students' preferences for different modelling domains?	FT	Educators
EI03	Do you have any suggestions for making modelling education more inclusive?	FT	Educators
Feedback → RQ4			
FE03	Do you have the feeling that your feedback can impact the quality of a course?	FT	Students
FE01	How do you assess and provide feedback on students' assignments?	FT	Educators
FE02	How do you evaluate the quality of your courses?	FT	Educators

We conducted a pilot study with three students and two educators from Germany, Iceland and Portugal, to help us evaluate the questionnaire clarity. The pilot participants were not part of the final study. Based on this feedback, we made minor adjustments to the wording of some questions and added one additional demographic

question. The final version of the questionnaire included four thematic categories based on the four RQs, each focusing on different aspects of participants' experiences with modelling assignments. These categories covered topics such as motivation, preferences and design choices of assignments, inclusion, and feedback.

Before participants completed the questionnaire, we explained the study's purpose and process. We informed them of the survey's voluntary nature, emphasising they could stop at any time and that all responses were anonymous. We did not require ethics approval since the study involved anonymous, non-sensitive questions, in accordance with the legal framework in Germany.

Table 1 shows the questionnaire for both students and educators.

Demographics. To better understand our participants, we collected basic demographic data. This included internal diversity factors such as age (*DE01*), gender (*DE02*), country of study or teaching (*DE04*), and whether they identified with a minority group, such as ethnic minorities or LGBTQ+ (*DE09*). We gathered information on the participants' educational and teaching experiences (*DE06,07,08,10,12*). Educators were asked if they are active members of the research community (*DE11*).

Motivation. To explore how problem domains influence motivation, we asked participants about their experiences with different modelling domains. The six domain categories (*MO01-01 to MO01-06*) were derived from an analysis of the authors' teaching materials as well as 31 papers from the EduSymp workshop at MODELS conference between 2019 and 2023 since it is a well-established venue that reflects current educational practice in modelling. We classified the domains based on the types of models students were required to create, the types of systems being modelled, and the tools used. For each year, one of the authors reviewed the selected papers to identify the model types, system types, and tools. All authors discussed the findings and grouped the domains into six final categories.

Preferences in Teaching Methods (Gamification, Collaboration). We aimed to identify the preferences of both students and educators regarding different teaching methods. Based on our analysis of teaching materials and the 31 EduSymp papers, we found that gamification is often suggested as a potential method (e.g., [8]). Additionally, it is closely linked to student motivation [53]. Before asking participants about their opinions on the use of gamification in modelling education (*PR02*), we provided a brief explanation of the concept to ensure understanding. We also explored preferences regarding collaboration in modelling assignments, asking participants whether they preferred working in groups, individually, or had no preference (*PR04*). We included this question based on our observations as teachers that group work can be a major (de-)motivator. To gain further insights, participants were encouraged to justify their choices through an open-ended question. Additionally, we asked educators for their perspectives on what motivates students to engage with modelling assignments (*PR06*).

Design of Assignments. We asked educators three open-ended questions about their approach to designing assignments. We inquired about how they select problem domains (*DA01*), which domains they typically use (*DA02*), and whether they take students' interests into account when designing assignments (*DA03*). These align with choosing socially relevant domains [38] and emphasize the benefits of allowing students to select their own topics [33].

Inclusion and Bias. Our analysis of teaching materials and the 31 EduSymp papers revealed that most approaches do not explicitly address diversity and inclusion aspects. To address this gap,

we asked students whether they feel excluded from certain topics (*IN02*), whether the choice of problem domains influences the inclusiveness of the course (*IN03*), and whether they believe these domains affect their learning success (*IN04*).

For educators, we asked how they consider diversity when designing assignments (*EI01*) and whether they have observed differences in student preferences based on diversity factors (*EI02*). We also asked both students and educators to suggest topics they believe would appeal to a diverse group of students (*IN01, EI03*).

Feedback. Feedback is crucial for improving teaching methods and assignments. We asked students whether their feedback had an impact on the quality of the course (*FE03*). We also asked educators how they assess feedback on student assignments (*FE01*) and how they incorporate this feedback into their teaching practices (*FE02*).

3.2 Data Collection and Participants

Our target group includes university students enrolled in computer science or related fields of study. To participate, students must have completed courses that covered software modelling to some extent, such as UML or Business Process modelling. Similarly, educators who have taught courses that included software modelling were invited to participate. The data collection took place between February and August 2025.

Educators. We sent an invitation to our educator survey to 117 educators via email in early February 2025. These invitees were selected from past program committees of the Models conference and authors of the EduSymp proceedings. Of the 117 invitations sent to professionals, 22 educators agreed to participate in the study, while 5 email addresses were undeliverable.

Students. We invited students from the authors' universities (i.e., Bayreuth, Darmstadt, Nova, Reykjavik) to fill in the student survey.

Overall, our dataset includes 112 participants, where 90 responses are from students and 22 from educators.

3.3 Data Analysis

To address our research questions, we used both quantitative and qualitative analysis methods.

3.3.1 Quantitative Analysis. For the closed-ended survey items, we conducted statistical analyses separately for students and educators. Ratios were calculated to provide descriptive insights into the distribution of responses. To test for statistically significant differences, we applied Fisher's exact test (Fisher) as implemented in R, using a significance level of $\alpha = 0.05$, as it is appropriate for small sample sizes and categorical data. We tested for differences in *MO01-01 to MO01-05* and in *PR02* between students and educators, by student gender, and by student minority status.

3.3.2 Qualitative Analysis. For the open-ended responses, we adopted content analysis [36] and thematic analysis [7] depending on the type of open-ended question. To support collaboration, we used a shared Google Sheets document to store all anonymised responses. Three researchers independently reviewed a subset of 20% of student and educator responses to agree on the suitability of the qualitative method. The remaining responses were divided among the researchers based on the research questions, who coded them individually. After completing the individual coding, we triangulated

the review process by having a second researcher review the coding. We resolved disagreements between researchers through structured discussion in joint online meetings, where they compared interpretations, justified their coding decisions with reference to the data, and refined code definitions until consensus was reached. This approach follows established practices in qualitative research that emphasise the collaborative construction of meaning rather than statistical agreement measures [7, 44].

We analysed factual or list-oriented responses (e.g., typical domains used) using *content analysis* and coded them into concrete categories. This includes the variables *DA02*, *EI02*, *FE01*, *FE02*, *FE03* (Table 1). In contrast, we analysed responses that provided reasoning, reflections, or broader perspectives (e.g., rationales for domain selection, considerations of diversity, or strategies to encourage student engagement) using *thematic analysis* to identify recurring patterns and overarching themes. This includes the following variables: *MO02*, *DA01*, *DA04*, *PR03*, *PR05*, *PR06*, *EI01*, *EI03*. This mixed strategy ensured that each variable was analysed with a method appropriate to the data's richness. Given the domain-specific nature of the questions and the relative consistency of responses, the dataset was considered sufficient to support meaningful qualitative interpretation without the aim of statistical representativeness.

3.4 Threats to Validity

Construct validity. A potential threat lies in the design of our survey instruments. While we based our questions on prior work, conducted a pilot study, and iteratively refined them, some constructs (e.g., inclusiveness) may not be fully captured through self-reported data. To mitigate this, we used both closed and open-ended questions and triangulated across students' and educators' perspectives.

Internal validity. Researcher bias in the qualitative coding process may influence responses interpretation, so three researchers independently coded a subset of responses, discussed disagreements, and agreed on a shared coding scheme before coding the full dataset.

Researcher positionality. The backgrounds of the research team may have shaped the framing of the study and the interpretation of data. Among the five authors, two identify as women and three as men; four are based in Europe and one in Asia. Three of the authors are active modelling educators, which provided valuable contextual knowledge but may also have introduced implicit assumptions about teaching practices. We sought to mitigate these influences by engaging in collaborative coding, reflexive discussions, and by making our analytic process transparent.

Conclusion validity. Our analyses may be affected by limitations in statistical power (e.g., small subgroups of participants) or the potential for Type I/II errors. We used appropriate statistical methods and complemented quantitative analyses with qualitative insights.

External validity. The generalisability of our results is limited by our sample of students and educators, which may not be representative of the broader population. We mitigated this by including participants from diverse institutions and backgrounds, and by clearly reporting demographic information.

4 Results

This section presents the findings for each research question. As the questionnaire was fully optional to align with ethical considerations,

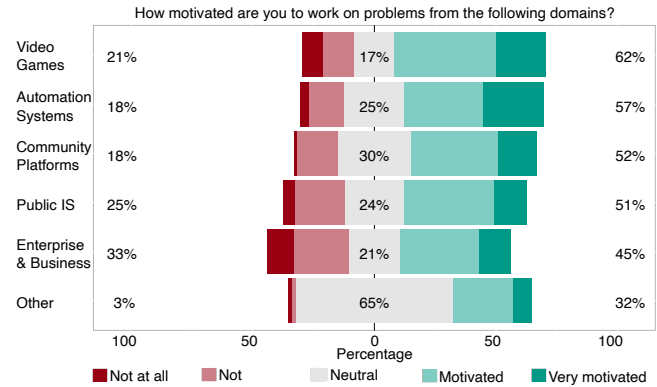


Figure 1: Students' motivation ratings by domain (RQ1).

totals across demographic, quantitative, and qualitative results may differ, as some participants chose not to answer all questions.

4.1 Demographics

4.1.1 Students. From students, we collected 90 valid answers (25 women, 55 men, 3 non-binary, 1 other and 6 did not answer). Of these, 62 had a bachelor's degree and 22 a master's degree. The majority ($n = 47$) had no relevant work experience, followed by 18 students with less than 1 year of experience and 17 students with 1 to 5 years of experience. Similarly, 49 students declared to have no software modelling experience, 19 students had less than 1 year of modelling experience, and 15 students had 1 to 5 years of experience. However, we recruited only students who had taken at least one university-level course containing modelling topics, ensuring some exposure to these topics. 41 students declared that they belonged to at least one minority: 18 students declared to belong to an ethnic minority, 13 declared to belong to LGBTQI+, 4 declared to be disabled, 5 to be socio-economically disadvantaged, and 11 students declared to belong to another minority.

4.1.2 Educators. We recruited 22 educators, 19 of whom indicated that they are also active in modelling research. The sample consisted of 17 men and 4 women (and 1 who did not answer *DE02*). One participant indicated belonging to a gender minority, and two indicated belonging to another minority (without a free-text description). The educators' ages covered a broad range from 25-34 ($n = 5$), 35-44 ($n = 7$), 45-54 ($n = 4$), to 55 or older ($n = 5$). Most courses (*DE10*) taught by educators in our sample relate to core SE principles such as Software Engineering, Requirements Engineering, Project Management, Software Modelling and Design (e.g., UML, BPMN, design patterns). Some focus on programming and implementation, like object-oriented programming, and domain-specific languages.

4.2 RQ1: Motivation

We present quantitative findings on domain preferences for both students and educators, followed by qualitative themes from each group, and conclude with a comparison of their perspectives.

4.2.1 Students Perspective. Figure 1 shows how motivating students find different problem domains (*MO01*). Interestingly, none

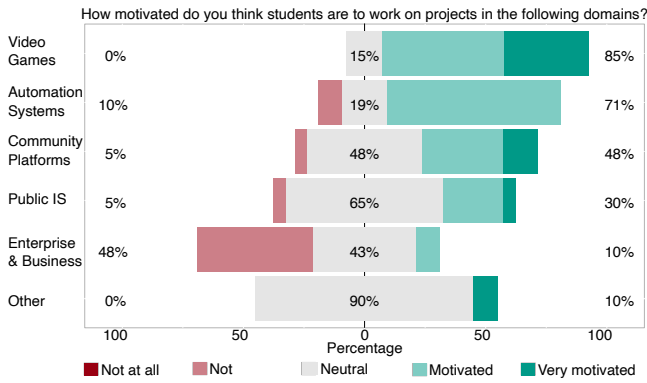


Figure 2: Educators' motivation ratings by domain (RQ1).

of the domains leans clearly in either direction, i.e., students have quite different opinions on what motivates them.

By comparing how motivating students of different subgroups find example domains, we see a statistically significant difference in how men and non-men view public information systems (Fisher, $p \approx 0.043$) and automation systems (Fisher, $p \approx 0.028$). Between minorities and non-minorities, there is a statistically significant difference for video games (Fisher, $p \approx 0.038$).

To better understand *why* student participants find certain domains motivating, we asked them to explain their responses in a free text field, which we analysed qualitatively.

The analysis of the students' answers for motivation (MO02) resulted in three major themes (shown *emphasised* in the following).

Personal Interests as Primary Drivers. Most students in our sample (49) explained that their motivation to work on projects depends on their own interests. They said their own interests are the main factor in deciding whether to work on a project that is related to certain domains. The content analysis revealed that the most mentioned topic here was *video games*. However, while most of the participants explained having specific interests in that domain, some explicitly reported not being interested.

Relevance of Subjects. Twelve students explained that their motivation to work on a subject is mainly driven by what they think is relevant for society, directly, or creates value to benefit society. Our content analysis did not reveal any specific outstanding domains in the qualitative answers.

Working in Social Environments. One student shared that they are interested in working within social settings. This shifts the emphasis away from the subject matter itself and toward the context in which the work takes place, highlighting a preference for socially engaging environments. It indicates a desire for interaction and collaboration with others throughout the projects.

4.2.2 Educators Perspective. Figure 2 shows which domains educators view as motivating for students. Most educators in our sample believe video games and automation systems are motivating for students. Students' answers are more polarised. Similarly, no educator in our sample believes that video games are demotivating, and only one believes that community platforms and public information

systems are demotivating for students. Two educators believe automation systems are demotivating. Interestingly, 10 of 22 educators believe business and enterprise systems are demotivating.

According to the qualitative analysis of the educators side (MO04), we found four themes.

Specific Students' Interests. Seven educators answered by explaining that students seem to be motivated by their interests and named specific domains that are the main interests of general students in their opinion/experience. Our content analysis showed that video game related words were more frequently present than others, even though there were no major differences.

Domain-based Students' Interests. Five educators also agreed that the students interests are the main driver but argued that the interests depend on the main domain of the students interests. Instead of naming concrete domains, those educators generally related aspects such as the study field or the general domains of interests to the motivation of students.

Real Subjects. One educator argued that students are interested in subjects that are real and complex. This suggests a need for real life examples that are not artificially created only for teaching purposes. The respective educator also explained to teach engineering students and argued that this might be a potential reason for this kind of real world focused motivation.

Own Knowledge. Another educator explained to have answered based on their own knowledge. Since they are not experienced with certain domains themselves they do not teach those.

4.2.3 Comparison of Students and Educators. We observe a statistical significant difference for students' and educators' views on public information systems (Fisher, $p \approx 0.018$), automation systems (Fisher, $p \approx 0.008$), and enterprise and business systems (Fisher, $p \approx 0.004$). Qualitatively, both groups agree that personal interest drives motivation, though educators in our sample tend to generalise about which domains interest students, while students emphasise diverse preferences. Educators also highlighted real-world relevance less often than might be expected, possibly because they are aware of the complexity such examples can introduce [13, 45].

RQ1 Summary. Students' interests is seen as an universal driver of motivation. Educators had strong views about which domains are motivating, whereas students' opinions vary widely.

4.3 RQ2: Teaching Methods and Designs

We investigated how preferences for *teaching methods* (PR02-PR06) and *assignment design choices* (DA01, DA02, DA04) affect motivation, from the perspective of students and educators.

4.3.1 Students Perspective.

Gamification (PR02-03). We asked students about the role of gamification as a motivator in modelling education.

A clear majority of students stated that gamification increases their motivation (PR02, $n = 40$ strongly agree, $n = 30$ agreed), with only 8 students stating that it does not affect their motivation and 4 that it decreases their motivation. There was no statistically significant difference in this question between men and non-men (Fisher, $p \approx 0.599$) as well as between students belonging to at least one minority and students belonging to none (Fisher, $p \approx 0.568$).

When asked to explain their views, students gave several reasons for and against gamification. Most students in our sample (37) value how gamification may *help with motivation*, and four refer to potential benefits in *increasing learning outcomes*. Some students (30) report just *liking gamification*. However, gamification can backfire as it can *lead to frustration* (5), four highlight that some students *dislike competition*, and three are concerned that gamification may *distract students from learning outcomes*. Three students express how gamification *feels condescending* to them.

Collaboration (PR04-PR05). We asked students about collaborative work in modelling assignments.

The picture for collaboration practices (PR04) is fairly balanced: 32 students stated that they prefer group projects, 29 preferred individual projects, and 21 students had no preference. There was no statistically significant gender difference in this questions.

When asked to explain their views, students gave several reasons for and against collaboration. The *balance between students who prefer to work in teams (32) and those who prefer to work solo (29)*, reflects on the weight those students attribute to the *benefits of discussing (11) or cross-evaluating their modelling attempts (4)*, as well as the *development of teamwork skills (3)*, versus concerns related to *distrusting their peers abilities and commitment to those projects (15)*, and *fears of uneven workload balance (4)*.

4.3.2 Educators Perspective. We also asked educators about gamification and collaboration (PR02-05), as well as other strategies they use to encourage engagement (PR06) and how they design assignments (DA01, DA02, DA04).

Gamification (PR02-03). Most educators in our sample see potential in *using gamification to motivate students*, but tend to take it with a grain of salt. Some educators (4) expressed concerns about *over-reliance on gamification*, whereas others (5) have no experience with it. A few educators (3) believe students like gamification, but others (2) highlight that it does not work for all students:

“I think gamification helps to motivate students, as long as there is not a strong competitive factor (e.g. public leaderboards). If there is, students have widely different reactions, thus affecting motivation differently depending on the student.” ED101

Collaboration (PR04-PR05). Most educators reported that they believe that *students prefer to work in groups (17)*. They identify *pedagogical benefits* to having students work in teams, as students get to *discuss their opinions* on the models they create (14) and *learn more* in the process. These discussions highlight that there is often more than one possible modelling solution to a given problem (6), which supports the argument for group work. *Scalability*, both in terms of modelling challenges and evaluation work by educators, favours collaborative teams. Educators recognise the benefits of including individual modelling assignments as a way of *fostering autonomy (2)* and *preventing uneven workload among students (1)*.

Encouraging students to model (PR06). We asked educators about encouraging students to engage with modelling assignments.

Some educators (5) make these assignments mandatory as *part of the evaluation process*. One educator suggested that a further motivation could be that the written exam topic could be the assigned

project. Three educators highlighted the relevance of using *interesting domains*, e.g., by allowing students to suggest the application domains they want to work on. A complementary approach is to highlight the modelling’s *relevance to industry*, e.g., by drawing on their own experience. Two educators prefer highlighting the different *modelling alternatives*, while two others stress the benefits of providing *personalised feedback* to students.

In addition, we asked educators how they select domains (DA01), which domains they choose (DA02), and how students’ interests influence this choice (DA04). These questions focused on educators as they are responsible for designing assignments.

Choosing modelling domains (DA01). Educators mention students’ *interest* as the most frequently cited criterion for selecting a domain for a modelling assignment (8). The domain’s *alignment with the student profile* is also considered important (5). One educator uses both as they mentioned to “*choose a domain that fits within their degree [...] and their interest to engage them as much as possible.*” ED86 General *relevance and suitability* are mentioned by six educators each. *Fitness to the learning goals* is mentioned by 4.

Used domains (DA02). There is a wide variety of specific domains in use, with banking (4) and university (4) being the most frequent. More broadly, educators mentioned domains related to travel and transports (11), organisational and business processes (10), education and knowledge (9), digital media and entertainment (9), cyber-physical systems (6), and e-commerce and shopping (2).

Student interest and motivation influence on modelling assignments design (DA04). The relevance educators assign to students’ interests and motivation is mixed. Some educators consider it important (7), or relatively important (3). Others find it not important (4). When considering students’ interests, some educators mention domain familiarity (3) and fitness to learning goals (3) as a criterion for their choice, followed by using a tangible domain (2).

4.3.3 Comparison of Students and Educators. Students believe that gamification increases their motivation more than educators do (Fisher, $p < 0.001$). Qualitatively, students are more enthusiastic about gamification overall, though both groups recognise it can backfire, particularly when it becomes overly competitive.

Regarding collaboration, there is a notable mismatch: most educators in our sample assume students prefer group work, yet students are evenly split in their preferences. Students who favour individual work often cite concerns about peers’ commitment and uneven workload, while educators emphasise the pedagogical benefits of discussion and learning from multiple perspectives.

RQ2 Summary. Gamification may increase motivation, but could backfire if poorly designed. Educators view collaboration more positively and select relatable domains to engage students.

4.4 RQ3: Inclusion and Bias

We investigated how students perceive inclusiveness in assignment domains (IN02–IN04) and how educators consider diversity and bias in domain selection (EI01–EI03).

4.4.1 Students Perspective.

Experiences of Exclusion (IN02). Five main themes emerged from students' experiences: no exclusion, exclusion occurred due to unfamiliar domains, representational issues (e.g., gender or queerness), assignment settings, and accessibility barriers.

Most students (39) reported no exclusionary experiences, e.g., due to the assignments having no connection to real life.

However, several students described exclusion linked to domain familiarity, e.g. if they had to build *"a project which included game elements and if you don't game then you were at a disadvantage."*^{ST5} This unfamiliarity can stem from culturally specific references and backgrounds when the assignment is

"based on a game that's well-known mainly in Europe. I even read through the entire Wikipedia article about the game rules, but I still couldn't fully understand it"^{ST75}

Other students (9) pointed to representational aspects:

"When thinking about the LGBTQ+ community... I can recall some assignments only referring to male and female, which could be a bit disregarding of other genders"^{ST19}

Further, students (3) highlighted not the topic but the assignment setting itself as exclusionary, e.g. if you prefer to work alone:

"A big group where work is conducted in class is pretty exclusionary to people with very big motor disabilities"^{ST32}

Impact of Domain Choice on Inclusiveness (IN03). Students highlighted four key themes: no impact (18), ambivalent opinions (12), and domain choice affects motivation depending on their background and problematic content (24).

One of the reasons why students think of the impact on inclusiveness is the lack of sense of belonging since if all

"assignments were to be specific to one type of person, others could feel demotivated and think this wasn't the course for them"^{ST28}

This is also due to the diversity in backgrounds as *"a topic that is familiar to some may be difficult for others."*^{ST54} Concerns about biased or problematic domains were also raised:

"In some domains there are inherited misogyny or phobia, in a modelling assignment it could be hard to get rid of them"^{ST51}

Impact on Learning Success (IN04). Students described three themes regarding learning outcomes (IN04): familiarity or interest in a domain improves understanding, unfamiliar domains can reduce engagement, and motivation mediates the effect. Most students (33) reported positive impact due to interesting domains:

"Familiar or interesting domains boost understanding, abstract or unfamiliar ones can hinder it. Helpful: Healthcare, transport, education. Harder: Physics simulations, finance models"^{ST57}

Some domains increased motivation (e.g., Minecraft, sports), but individual preferences varied. Most students emphasised real-world relevance since if the domain is close to their life, *"it could be interesting to look at familiar things from a new perspective."*^{ST51}

Ten students felt there was little to no impact at all if the assignment is well-designed and explained since they *"try to understand the underlying concepts than the specific application"*^{ST42}

Preferred Domains for Diverse Groups (IN01). Three themes emerged regarding preferred domains (IN01): students favour societal relevance, everyday experiences that all students can relate to, and the ability to choose or personalise domains.

Students suggested broadly accessible and socially relevant domains, including societal topics (culture, mental health, climate; 19), community platforms (12), education (9), everyday student life (5), games which are not related to *gamer culture* (3).

Since there is often the risk of *othering* students who belong to an under-represented group, students emphasised choice:

"I think that's really hard as we are all different. I believe that the approach 'choose your own topic/theme' is the best one."^{ST28}

4.4.2 Educators Perspective.

Considering Diversity in Domain Selection (EI01). Educators' practices (EI01) revealed three main patterns: some explicitly consider diversity to increase accessibility and challenge stereotypes (7), others implicitly account for diversity via familiar contexts or student autonomy (4), and some do not consider diversity at all (5).

When explicitly considering diversity, strategies included using familiar domains for international students, avoiding conflict-prone domains, and selecting domains that challenge binary thinking.

When addressing implicit diversity, educators draw on everyday contexts relatable to all, use study-related domains, or allow students to choose. As one third do not consider diversity, inclusive assignment design is unevenly integrated into teaching practice.

Observed Differences in Student Preferences (EI02). Three themes emerged (EI02): educators who notice differences adjust for personal interests or background (3), some acknowledge potential differences but do not act on them (2), and most are unaware of differences or bias (11). For example, one observed gender-specific patterns:

"Some games appeal more to a male audience, e.g., a spaceship strategy game. Yet, most games appeal to all audiences. [...] To be more inclusive towards women in computer science education, a colleague argued that projects for sharing, ecological or sociological improvements are more appealing to women"^{ED152}

Suggestions for Inclusive Modelling Education (EI03). Four themes guided educators' suggestions (EI03): providing a variety of domains and task types (6), fostering interaction with diverse perspectives (2), using accessible tools, and considering student backgrounds.

The most common recommendation is the choice of task and variety of domains, yet, apply it with caution:

"But for certain physical disabilities (visual impairment, physical impairment of hands) modelling is quite hard because of the tooling not adapted to these students."^{ED70}

Another suggested a gradual sequence of assignments:

“We need small, comprehensive examples from the student’s life or experience, an inclusive medium example to train, and finally a real-world example. Only the small and medium case must be inclusive, whereas the big one should prepare the students for their job” ED152

Two educators emphasised interaction and engagement with diverse perspectives, suggesting, for instance:

“Having students interact with people (or at least personas) of disadvantage categories as ‘clients’ of their projects” ED95

Considering students’ background (e.g., cultural, educational, socio-economic), educators warned against unfamiliar domains:

“Try not to pick domains that you yourself like, but students may not know/like, or which may even discriminate. Classical example: Booking a hotel – some students may never have stayed in a hotel since they cannot afford it” ED143

Two educators were sceptical about addressing inclusivity at course level, seeing it as a broader institutional responsibility.

4.4.3 Comparison of Students and Educators. Students and educators in our sample agree that interest influences motivation and inclusiveness. However, students prioritise choice and societal relevance, while educators focus more on familiarity and manageability of domains. Students more clearly identify exclusion risks (e.g., unfamiliar references, economic assumptions, limited representation), whereas many educators are unaware of such issues or do not actively address them, suggesting a misalignment between intentions and students’ lived experiences.

RQ3 Summary. Most students in our sample find assignment domains mainly inclusive. Students value choice and societal relevance, while educators’ awareness and strategies vary.

4.5 RQ4: Feedback

We investigated how students perceive feedback (FE03), how educators provide it (FE01), and how they evaluate their course quality (FE02) to explore the dynamics of feedback in modelling education.

4.5.1 Students Perspective. Students’ responses (FE03) clustered around three major themes: perceived positive impact, conditional impact, and lack of visible impact. Interestingly, they all interpreted feedback as course quality feedback, not on their learning.

Positive Impact. Most students (32) believed feedback enhances course quality, emphasising motivational and practical benefits:

“Without any feedback, the teaching person will think that everything is perfect and not change anything. Feedback might motivate to change something and enhance the quality of the course.” ST41

The positive impact allows also for hearing diverse views since *“many opinions might give new perspectives”*. ST56

Conditional Impact. Several students (19) expressed more nuanced views by linking the effectiveness to timing, lecturer responsiveness, and the visibility of integration and peer agreement:

“it can also become frustrating when 1. feedback is not taken at a point when it can still improve the students taking the course at that moment, and 2. when we can see that feedback from previous years was clearly not taken into consideration, as some issues remain recurring.” ST35

Some students felt their input mattered only if peers agreed or if surveys were representative. In addition, some stressed that impact depends on personal fit:

“My feedback would improve the quality for me. If none share my opinion it’d probably decrease.” ST28

Lack of Impact. Several students (13) doubted that feedback influenced teaching, pointing to entrenched course structures or low institutional priority of teaching:

“Teaching is just a side task for the teachers and professors. Normally they just stick to the old course structures and slides.” ST51

4.5.2 Educators Perspective. Educators reported three main approaches for their practices (FE01, FE02).

Formative and personalised feedback. Educators (6) emphasised one-on-one discussions and or small-group feedback, often provided in consultation hours or practical sessions:

“I periodically hold consultation sessions where students can discuss their group assignments. Practical sessions are also used to provide feedback as needed.” ED07

Scalable mechanisms. Several educators (5) highlighted iterative cycles, peer review, or automated platforms, particularly for larger cohorts. One approach combined automation and peer processes:

“Small single-person assignments receive automatic and individual feedback via an exercise platform. For the group project, we employ a double-blind peer-review process, where each group evaluates the models of two other groups.” ED18

Course evaluation. Educators (12) typically relied on standardised student surveys and university-wide processes to assess quality. Some highlighted long-term course refinement.

4.5.3 Comparison of Students and Educators. Both groups agree that feedback is important, but the perception of its effectiveness differs. While educators actively provide feedback through multiple channels, students sometimes perceive it as ineffective if not visibly acted upon. This gap between provision and perceived impact highlights a potential risk: students may feel their voices do not matter, especially those from under-represented groups.

RQ4 Summary. Students value feedback mainly when visibly acted upon. Educators provide it regularly, but perception gaps can reduce its motivational impact.

Table 2: Recommendations for software modelling education.

Goal	Recommendation
Use students' interests	Offer opportunities to choose the project domains by students, if possible (<i>RQ1</i> , <i>RQ2</i>).
Show societal relevance	Link domains to real-world, socially meaningful problems, not just technical examples (<i>RQ1</i> , <i>RQ3</i>).
Use gamification wisely	Keep it engaging but avoid overemphasis on competition (<i>RQ2</i>).
Support collaboration	Scaffold teamwork, discuss <i>fair</i> work distribution, and use mechanisms like blind peer evaluation (<i>RQ2</i>).
Avoid narrow examples	Do not rely on culturally specific or stereotypical domains; prefer everyday contexts (<i>RQ1</i> , <i>RQ3</i>).
Make feedback visible	Collect feedback iteratively and show students how their input influences teaching, giving minorities a voice (<i>RQ4</i>).

5 Discussion

Our study provides insights into students' motivations in software modelling courses, highlighting how domain diversity, the inclusiveness of assignments, and the feedback they receive jointly influence student engagement.

5.1 Synthesis of Results

Our findings suggest a mismatch between educators' and students' perceptions of what makes modelling motivating. Educators hold more uniform and narrow views on suitable domains (*RQ1*) and teaching methods such as gamification or collaboration (*RQ2*). In contrast, students report a wider variety of interests and are cautious about approaches that feel forced or overly competitive. What consistently drives motivation is when assignments connect to students' personal interests, or to socially meaningful, everyday domains, which resonates with prior work on autonomy and relevance in learning [12, 54, 57]. This reframes inclusiveness beyond technical access: while most students report no overt exclusion (*RQ3*), those who do point to culturally specific references and economic assumptions in domain choices educators do not anticipate, connecting inclusivity concerns (*RQ3*) to domain and teaching method selection practices explored in *RQ1* and *RQ2*. Students feel more included when domains reflect their lived realities, echoing calls for diversity-rich curricula [11, 49]. The feedback gaps identified in *RQ4* may partially explain the persistent educator-student misalignments observed across *RQ1-3*: while educators design structured mechanisms, students value it only when they can see tangible effects on their learning, supporting earlier observations on the importance of formative, responsive practices [13, 35].

Inclusive modelling education depends on creating assignments that make students feel represented, not just technically competent, in line with findings from Computer Science education [43, 56]. It requires educators to view inclusiveness as more than access, instead designing tasks that acknowledge students' voices, backgrounds, and aspirations. Overall, our findings highlight that effective modelling education must balance educator intentions with student experiences, leveraging domain diversity, inclusive practices, and timely feedback to increase both motivation and engagement.

5.2 Connecting Educators and Students

Table 2 summarises our practical recommendations for more inclusive software modelling education.

While demographic factors are often reported, the diversity of domains students engage with remains under-explored [21]. Our results show that engagement is strongly shaped by individual domain preferences, with many students motivated by socially relevant problems that educators often overlook (*RQ1*, *RQ3*). This suggests that educators should offer choice and link tasks to real-world issues rather than relying on assumedly motivating technical topics, echoing prior work on "diversity in problem-solving" [31].

Gamification is another challenge: most students in our sample enjoy it, but some find competitive elements discouraging (*RQ2*), in line with earlier findings [29]. Educators should use a broader set of game elements to enhance engagement, ideally integrated with collaborative work. Collaboration itself divides opinions: while educators highlight its benefits, some students distrust peers or fear uneven workload distribution (*RQ2*), a pattern that resonates with earlier studies [32]. This points to a need for better communication and scaffolding of collaborative tasks.

Students value course feedback when they see that educators act upon it (*RQ4*). End-of-semester surveys rarely demonstrate impact and may silence minority perspectives. Iterative feedback with visible follow-up during courses can address this gap. By offering autonomy in domain choice, applying gamification thoughtfully, scaffolding collaboration, and ensuring responsive feedback, educators can foster more inclusive and motivating modelling education.

6 Conclusions and Future Work

Our study shows that motivation, domain choice, inclusivity, and feedback are interconnected in software modelling courses. From a students perspective, socially relevant, relatable, and self-selected domains as well as well-executed gamification enhance engagement, while visible and responsive feedback is essential for inclusive participation. Educators' assumptions about student interests and reliance on standardised surveys can limit motivation and inclusion.

This is the first empirical study comparing educators' and students' perspectives on these factors in modelling education. Our findings are based on self-reported perspectives, which are important indicators related to motivation. However, actual learning might be affected by further factors. Intervention studies should therefore test whether incorporating our recommendations *actually* improves student engagement and outcomes, particularly for underrepresented groups. Finally, cross-cultural replication helps determine whether our findings from predominantly European contexts generalise to other educational settings, or whether culturally-specific approaches to inclusive modelling education are needed.

Acknowledgments

This study was developed as a part of the GI-Dagstuhl Seminar 23473: Human Factors in Model-Driven Engineering. We thank Schloss Dagstuhl for the opportunity to conduct the seminar. The work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 414984028 – SFB 1404 FONDA, and NOVA LINCS (UID/04516/2025) with the financial support of FCT/IP.

Data Availability

We provide the anonymised survey data, including questionnaires and introductory text, in our replication package [26].

References

- [1] Luciane TW Agner, Timothy C Lethbridge, and Inali W Soares. 2019. Student experience with software modeling tools. *Software & Systems Modeling* 18, 5 (2019), 3025–3047. <https://doi.org/10.1007/s10270-018-00709-6>
- [2] Seiko Akayama, Birgit Demuth, Timothy C Lethbridge, Marion Scholz, Perdita Stevens, and Dave R Stikkolorum. 2013. Tool Use in Software Modelling Education. In *EduSymp@ MoDELS 2013*. CEUR, Miami, USA, 1–22. <https://ceur-ws.org/Vol-1134/paper6.pdf>
- [3] Pasquale Ardimento, Mario Luca Bernardi, Marta Cimitile, and Michele Scalera. 2024. Enhancing Software Modeling Learning with AI-Powered Scaffolding. In *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems* (Linz, Austria). ACM, New York, NY, USA, 103–106. <https://doi.org/10.1145/3652620.3687776>
- [4] Jean-Christophe Bach, Antoine Beugnard, Jean-Loup Castaigne, Julien Mallet, Salvador Martinez, and Maria-Teresa Segarra. 2022. An analysis of software design understanding & motivation of engineering students. In *Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings* (Montreal, Quebec, Canada). ACM, New York, NY, USA, 123–130. <https://doi.org/10.1145/3550356.3556506>
- [5] Omar Badreddin, Timothy Lethbridge, Arnon Sturm, Waylon Dixon, Abdelwahab Hamou-Lhadj, and Ryan Simmons. 2015. The effects of education on students' perception of modeling in software engineering. In *Proceedings of the First International Workshop on Human Factors in Modeling (HuFaMo 2015)* (Ottawa, Canada). CEUR, Leiden, Netherlands, 39–46. <https://ceur-ws.org/Vol-1522/Badreddin2015HuFaMo.pdf>
- [6] Dominik Bork, Stefan Klikovits, Judith Michael, Lukas Netz, and Bernhard Rumpe. 2025. Inclusive Model-Driven Engineering for Accessible Software. In *MODELS New Ideas and Emerging Results (NIER) Track 2025* (Grand Rapids, MI, USA). IEEE, New York, NY, USA, 253–259. <https://doi.org/10.1109/MODELS67397.2025.00030>
- [7] Virginia Braun and Victoria Clarke. 2022. Conceptual and Design Thinking for Thematic Analysis. *Qualitative Psychology* 9, 1 (2022), 3.
- [8] Antonio Bucchiarone, Antonio Cicchetti, Simone Bassanelli, and Annapaola Marconi. 2021. How to merge gamification efforts for programming and modelling: a tool implementation perspective. In *2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)* (Grand Rapids, MI, USA). IEEE, New York, NY, USA, 721–726. <https://doi.org/10.1109/MODELS-C53483.2021.00116>
- [9] Antonio Bucchiarone, Maxime Savary-Leblanc, Xavier Le Pallec, Antonio Cicchetti, Sébastien Gérard, Simone Bassanelli, Federica Gini, and Annapaola Marconi. 2023. Gamifying model-based engineering: the PapyGame experience. *Software and Systems Modeling* 22, 4 (2023), 1369–1389. <https://doi.org/10.1007/s10270-023-01091-8>
- [10] Patrick Buckley and Elaine Doyle. 2016. Gamification and student motivation. *Interactive learning environments* 24, 6 (2016), 1162–1175. <https://doi.org/10.1080/10494820.2014.964263>
- [11] Sheryl Burgstahler. 2009. *Universal Design of Instruction (UDI): Definition, Principles, Guidelines, and Examples*. Technical Report. University of Washington, DO-IT.
- [12] Mustafa Cevikbas, Gilbert Greefrath, and Hans-Stefan Siller. 2023. Advantages and challenges of using digital technologies in mathematical modelling education—a descriptive systematic literature review. *Frontiers in Education* 8 (2023), 1–17. <https://doi.org/10.3389/feduc.2023.1142556>
- [13] Shalini Chakraborty and Grischa Liebel. 2023. We do not understand what it says—studying student perceptions of software modelling. *Empirical Software Engineering* 28, 6 (2023), 149. <https://doi.org/10.1007/s10664-023-10404-w>
- [14] LaVar J Charleston, Sherri Ann Charleston, and Jerlando F L Jackson. 2025. Using Culturally Responsive Practices to Broaden Participation in the Educational Pipeline: Addressing the Unfinished Business of Brown in the Field of Computing Sciences. *The Journal of Negro Education* 83, 3 (2025), 400–419. <https://muse.jhu.edu/article/802984>
- [15] Kua Chen, Boqi Chen, Yujing Yang, Gunter Mussbacher, and Dániel Varró. 2024. Embedding-based Automated Assessment of Domain Models. In *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*. ACM, New York, NY, USA, 87–94. <https://doi.org/10.1145/3652620.3687774>
- [16] Stanislav Chren, Barbora Buhnova, Martin Macak, Lukas Daubner, and Bruno Rossi. 2019. Mistakes in UML diagrams: analysis of student projects in a software engineering course. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*. IEEE, New York, NY, USA, 100–109. <https://doi.org/10.1109/ICSE-SEET.2019.00019>
- [17] Federico Ciccozzi, Ivano Malavolta, and Bran Selic. 2019. Execution of UML models: a systematic review of research and practice. *Software & Systems Modeling* 18, 3 (2019), 2313–2360. <https://doi.org/10.1007/s10270-018-0675-4>
- [18] Peter J Clarke and Alfonso Pierantonio. 2018. Teaching modeling: a software perspective. *Computer Science Education* 28, 1 (2018), 1–4. <https://doi.org/10.1080/08993408.2018.1486535>
- [19] Anthony J Cowling. 2003. Modelling: a neglected feature in the software engineering curriculum. In *Proceedings 16th Conference on Software Engineering Education and Training, 2003.(CSEE&T 2003)*. IEEE, New York, NY, USA, 206–215. <https://doi.org/10.1109/CSEE.2003.1191378>
- [20] Anthony J Cowling. 2005. The role of modelling in the software engineering curriculum. *Journal of Systems and Software* 75, 1-2 (2005), 41–53. <https://doi.org/10.1016/j.jss.2004.02.021>
- [21] Riya Dutta, Diego Elias Costa, Emad Shihab, and Tanja Tajmel. 2023. Diversity awareness in software engineering participant research. In *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. IEEE, New York, NY, USA, 120–131. <https://doi.org/10.1109/ICSE-SEIS58686.2023.00017>
- [22] Michel Estefan, Jesse Cordes Selbin, and Sarah Macdonald. 2023. From inclusive to equitable pedagogy: How to design course assignments and learning activities that address structural inequalities. *Teaching Sociology* 51, 3 (2023), 262–274. <https://doi.org/10.1177/0092055X231174515>
- [23] Mario Fuksa and Steffen Becker. 2021. Mini programming worlds: Teaching mdsd via the hamster simulator. In *2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*. IEEE, New York, NY, USA, 696–701. <https://doi.org/10.1109/MODELS-C53483.2021.00113>
- [24] Giacomo Garaccione, Riccardo Coppola, Luca Ardito, and Marco Torchiano. 2024. Gamification of business process modeling education: an experimental analysis. *Software and Systems Modeling* 23, 6 (2024), 1569–1594. <https://doi.org/10.1007/s10270-024-01171-3>
- [25] Martin Gogolla and Perdita Stevens. 2018. Teaching modeling in computer science as an ecosystem: a provocative analogy. *Computer Science Education* 28, 1 (2018), 5–22. <https://doi.org/10.1080/08993408.2018.1463634>
- [26] Isabella Graßl, Christopher Lazik, Shalini Chakraborty, Grischa Liebel, and Miguel Goulão. 2025. <https://figshare.com/s/55ebff3020d2a2dd1d7c>.
- [27] Markus Hamann, Sebastian Götz, and Uwe Abmann. 2024. Towards an Interoperable Model-driven Automated Assessment System for Computer Science Education. In *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems* (Linz, Austria). ACM, New York, NY, USA, 95–102. <https://doi.org/10.1145/3652620.3687775>
- [28] Feifei Han, Robert A. Ellis, and Abelardo Pardo. 2022. The Descriptive Features and Quantitative Aspects of Students' Observed Online Learning: How Are They Related to Self-Reported Perceptions and Learning Outcomes? *IEEE Transactions on Learning Technologies* 15, 1 (Feb. 2022), 32–41. <https://doi.org/10.1109/TLT.2022.3153001>
- [29] Michael D Hanus and Jesse Fox. 2015. Assessing the effects of gamification in the classroom: A longitudinal study on intrinsic motivation, social comparison, satisfaction, effort, and academic performance. *Computers & education* 80 (2015), 152–161. <https://doi.org/10.1016/j.compedu.2014.08.019>
- [30] John Hutchinson, Jon Whittle, Mark Rouncefield, and Steinar Kristoffersen. 2011. Empirical assessment of MDE in industry. In *Proceedings of the 33rd international conference on software engineering* (Waikiki, Honolulu, HI, USA). ACM, New York, NY, USA, 471–480. <https://doi.org/10.1145/1985793.1985858>
- [31] Sonja M Hyrynsalmi. 2025. Challenges and opportunities: Implementing diversity and inclusion in software engineering university level education in Finland. *Journal of Systems and Software* 219 (2025), 112239. <https://doi.org/10.1016/j.jss.2024.112239>
- [32] Claudia Iacob and Shamal Faily. 2019. Exploring the gap between the student expectations and the reality of teamwork in undergraduate software engineering group projects. *Journal of Systems and Software* 157 (2019), 110393. <https://doi.org/10.1016/j.jss.2019.110393>
- [33] Ayaan M Kazerouni, Zainab Agha, Aleata Hubbard Cheuoua, Melissa Lee, Jane Lehr, Ilmi Yoon, and Zoë Wood. 2025. What Topic Domains Interest Students in Socially Responsible Computing Coursework?. In *Proceedings of the 25th Koli Calling International Conference on Computing Education Research*. ACM, New York, NY, USA, 1–7. <https://doi.org/10.1145/3769994.3770030>
- [34] Azad Khandoker, Sabine Sint, Manuel Wimmer, and Klaus Zeman. 2023. An interdisciplinary course on model-based systems engineering. In *2023 ACM/IEEE international conference on model driven engineering languages and systems companion (MODELS-C)*. IEEE, New York, NY, USA, 102–109. <https://doi.org/10.1109/MODELS-C59198.2023.00033>
- [35] Jörg Kienzle, Steffen Zschaler, William Barnett, Timur Sağlam, Antonio Bucchiarone, Silvia Abrahão, Eugene Syriani, Dimitris Kolovos, Timothy Lethbridge, Sadaf Mustafiz, et al. 2024. Requirements for modelling tools for teaching. *Software and Systems Modeling* 23, 5 (2024), 1055–1073. <https://doi.org/10.1007/s10270-024-01192-y>
- [36] Klaus Krippendorff. 2018. *Content analysis: An introduction to its methodology*. Sage publications, Thousand Oaks, CA, USA.
- [37] Gloria Ladson-Billings. 1995. But that's just good teaching! The case for culturally relevant pedagogy. *Theory into practice* 34, 3 (1995), 159–165. <https://doi.org/10.1080/00993408.2018.1486535>

- 1080/00405849509543675
- [38] Lucas Layman, Laurie Williams, and Kelli Slaten. 2007. Note to self: make assignments meaningful. In *Proceedings of the 38th SIGCSE technical symposium on Computer science education* (Covington, Kentucky, USA) (SIGCSE '07). ACM, New York, NY, USA, 459–463. <https://doi.org/10.1145/1227310.1227466>
 - [39] Grisca Liebel, Rogardt Haldal, and Jan-Philipp Steghöfer. 2016. Impact of the use of industrial modelling tools on modelling education. In *2016 IEEE 29th International conference on software engineering education and training (CSEET)*. IEEE, New York, NY, USA, 18–27. <https://doi.org/10.1109/CSEET.2016.18>
 - [40] Grisca Liebel, Jil Klünder, Regina Hebig, Christopher Lazik, Inês Nunes, Isabella Graßl, Jan-Philipp Steghöfer, Joeri Exelmans, Julian Oertel, Kai Marquardt, et al. 2024. Human factors in model-driven engineering: future research goals and initiatives for MDE. *Software and Systems Modeling* 23, 4 (2024), 801–819. <https://doi.org/10.1007/s10270-024-01188-8>
 - [41] Pierre Maier and Tobias Schwarz. 2024. UML++: Enhancing Student Learning of Object-Oriented Modeling through Executable Objects. In *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems* (Linz, Austria). ACM, New York, NY, USA, 107–114. <https://doi.org/10.1109/MODELS-C53483.2021.00113>
 - [42] Misaki Maruyama, Shinpei Ogata, Kozo Okano, and Mizue Kayama. 2018. Support tool for refining conceptual model in collaborative learning. In *Joint Conference on Knowledge-Based Software Engineering*. Springer, Cham, 147–157. https://doi.org/10.1007/978-3-319-97679-2_15
 - [43] Joseph E. Michaelis and David Weintrop. 2022. Interest Development Theory in Computing Education: A Framework and Toolkit for Researchers and Designers. *ACM Trans. Comput. Educ.* 22, 4, Article 43 (Dec. 2022), 27 pages. <https://doi.org/10.1145/3487054>
 - [44] Cliodhna O'Connor and Helene Joffe. 2020. Intercoder Reliability in Qualitative Research: Debates and Practical Guidelines. *International Journal of Qualitative Methods* 19 (Jan. 2020), 1609406919899220. <https://doi.org/10.1177/1609406919899220>
 - [45] Richard F Paige, Fiona AC Polack, Dimitrios S Kolovos, Louis M Rose, Nicholas Drivalos Matragkas, and James R Williams. 2014. Bad Modelling Teaching Practices. In *EduSymp@ MODELS*. CEUR, Leiden, Netherlands, 1–12. https://ceur-ws.org/Vol-1346/edusymp2014_paper_keynote.pdf
 - [46] Abelardo Pardo, Feifei Han, and Robert A. Ellis. 2016. Combining University Student Self-Regulated Learning Indicators and Engagement with Online Learning Events to Predict Academic Performance. *IEEE Transactions on Learning Technologies* 10, 1 (2016), 82–92. <https://doi.org/10.1109/TLT.2016.2639508>
 - [47] Beatriz Pérez and Ángel L Rubio. 2020. A project-based learning approach for enhancing learning skills and motivation in software engineering. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. ACM, New York, NY, USA, 309–315. <https://doi.org/10.1145/3328778.3366891>
 - [48] Marian Petre. 2013. UML in practice. In *2013 35th international conference on software engineering (icse)*. IEEE, New York, NY, USA, 722–731. <https://doi.org/10.1109/ICSE.2013.6606618>
 - [49] Siema Ramdas, Marieke Slootman, and Karen van der Zee. 2025. Creating equitable learning environments by building on differences in higher education: design and implementation of the MIXED model. *Learning Environments Research* 28 (2025), 307–322. <https://doi.org/10.1007/s10984-025-09524-5>
 - [50] Kimberly A. Scott, Kimberly M. Sheridan, and Kevin Clark. 2015. Culturally Responsive Computing: A Theory Revisited. *Learning, Media and Technology* 40, 4 (2015), 412–436. <https://doi.org/10.1080/17439884.2014.924966>
 - [51] Martina Seidl, Marion Scholz, Christian Huemer, and Gerti Kappel. 2015. *UML@ classroom*. Springer. <https://doi.org/10.1007/978-3-319-12742-2>
 - [52] Robert Szmurlo and Michał Śmialek. 2006. Teaching software modeling in a simulated project environment. In *International Conference on Model Driven Engineering Languages and Systems*. Springer, 301–310. https://doi.org/10.1007/978-3-540-69489-2_37
 - [53] Simone Tönhão, Marcelo Shigenaga, Julio Herculan, Andressa Medeiros, Aline Amaral, Williamson Silva, Thelma Colanzi, and Igor Steinmacher. 2023. Gamification in software engineering education: A tertiary study. In *Proceedings of the XXXVII Brazilian Symposium on Software Engineering*. ACM, New York, NY, USA, 358–367. <https://doi.org/10.1145/3613372.3614193>
 - [54] Kyle F Trenshaw, Renata A Revelo, Katherine A Earl, and Geoffrey L Herman. 2016. Using self-determination theory principles to promote engineering students' intrinsic motivation to learn. *International Journal of Engineering Education* 32, 3 (2016), 1194–1207.
 - [55] Naomi Unkelos-Shpigel, Julia Sheidin, and Moran Kupfer. 2019. Climb Your Way to the Model: Teaching UML to Software Engineering Students: Teaching Case. In *International Conference on Advanced Information Systems Engineering*. Springer, 40–46. https://doi.org/10.1007/978-3-030-20948-3_4
 - [56] Elizabeth von Briesen, Richard Dutton, Shannon Duvall, Duke Hutchings, Ryan Mattfeld, and Scott Spurlock. 2025. Interventions for Increasing Belonging and Inclusion in Undergraduate Computer Science Classrooms. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1* (Pittsburgh, PA, USA) (SIGCSETS 2025). ACM, New York, NY, USA, 1162–1168. <https://doi.org/10.1145/3641554.3701941>
 - [57] Jason Lee Weber, Barbara Martinez Neda, Sergio Gago-Masague, and Jennifer Wong-Ma. 2024. Maximizing Individual Learning Goals Through Customized Student-Project Matching (SPM) in CS Capstone Projects. In *2024 IEEE Frontiers in Education Conference (FIE)*. IEEE, New York, NY, USA, 1–8. <https://doi.org/10.1109/FIE61694.2024.10893018>
 - [58] Jon Whittle, Christopher N Bull, Jaejoon Lee, and Gerald Kotonya. 2014. Teaching in a software design studio: Implications for modeling education. In *MODELS Educators Symposium 2014*. CEUR, Leiden, Netherlands, 12–21. http://ceur-ws.org/Vol-1346/edusymp2014_paper_1.pdf
 - [59] Jon Whittle, John Hutchinson, and Mark Rouncefield. 2013. The state of practice in model-driven engineering. *IEEE software* 31, 3 (2013), 79–85. <https://doi.org/10.1109/MS.2013.65>