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

Anonymous Author(s)
University
City, Country

## Abstract

Student engagement is critical for effective learning in software modelling, yet fostering motivation and inclusivity remains a challenge. While existing research has primarily 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 shapes students' learning experiences. This study explores how problem domains and instructional design influence motivation, engagement, perceptions of inclusiveness, and the role of feedback in modelling education. To investigate these dimensions, we conducted parallel surveys of students and educators with 112 participants. Our findings reveal major 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. We provide practical recommendations for more inclusive modelling education through student-centred domain selection.

## CCS Concepts

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

## Keywords

Diversity, Modelling, Software Engineering Education.

**ACM Reference Format:**
Anonymous Author(s). 2025. Domain Diversity, Motivation, and Inclusion in Software Modelling Education. In *Proceedings of 48th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET '26)*. ACM, New York, NY, USA, 12 pages. https://doi.org/XXXXXXX.XXXXXXX

## 1 Introduction

Software modelling is a fundamental skill in software engineering (SE) education, equipping students with essential capabilities in abstraction, conceptualization, and system design [19]. As software systems grow increasingly complex, driven by diverse stakeholder requirements, varied user needs, and deployment across heterogeneous environments, models provide the critical abstractions needed to understand, design, and manage such complexity [26]. Despite its importance, software modelling has often been underemphasised in curricula [18]. This neglect poses challenges for both students and educators. For students, modelling can be conceptually demanding [12], with difficult tooling choices [1, 31] and a perception that it is detached from practical applications [6].

In contrast to many other areas of SE, modelling is often taught in a top-down and formal way [46]. Therefore, motivating the use of models and maintaining student engagement can be challenging.

One way to engage and motivate students in modelling education could be through an appropriate choice of example domains in assignments and lectures. Paige et al. [35] observe that *"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 example domains is therefore not merely illustrative, but can play a pivotal role in sustaining engagement. Fun, relatable, and varied domains can help students connect abstract modelling concepts to contexts they understand and value. However, many traditional modelling examples, such as library management systems or simple game development scenarios may fail to resonate with all students. This challenge is compounded in large, diverse classrooms, where students come from varied cultural backgrounds, educational experiences, and have varied preferences and professional aspirations [40].

Recent work calls for considering human factors in software modelling [32] and for considering inclusion and accessibility aspects in teaching modelling [7]. However, while there is substantial research on software modelling education with respect to factors such as tool use [1, 2, 31], student learning [3, 5, 33], assessment practices [14, 24], or specific classroom experiences [22, 28], there exists to our knowledge no work that investigates the choice of problem domains and how it influences student motivation. To address this gap, we answer 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 among educators experienced in modelling education. To mirror their views, we complemented this with a second survey among students with modelling experience. Based on answers from 22 educators and 90 students, our findings show that there is clear disagreement among


*ICSE-SEET '26, Rio de Janeiro, Brazil*

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/XXXXXXX.XXXXXXX

students which problem domains are motivating. While educators hold strong assumptions about which domains motivate students, students show much more diverse preferences driven primarily by personal interests and social relevance.

## 2 Background and Related Work

This section reviews prior work on software modelling education, student motivation factors, and inclusive teaching strategies for diverse problem domains.

### 2.1 Software Modelling Education

Research indicates that while modelling is part of many software engineering curricula, pedagogical emphasis often remains superficial, focused on syntax rather than conceptual modelling and model-driven thinking. Whittle et al. [47] 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 [17]. Gogolla and Stevens [23] conceptualize modelling education as an ecosystem, first identifying the factors that constitute this ecosystem and then describing the relationships among them. 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, 31]. Recent guidelines emphasise the need for teaching-specific modelling tools with easy setup, low friction, and meaningful feedback [29]. The field's reliance on UML as the dominant modelling language is well-established in Seidl et al.'s [41] instructional methodology. However, Petre [38] 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. [16], whose analysis of executable UML models reveals significant limitations in translating educational models to production systems. Empirical studies further document student struggles with fundamental modelling concepts like abstraction, multiplicity, and mapping from textual requirements to diagrams, highlighting prevalent misconceptions that hinder learning unless actively addressed [15]. Collectively, these challenges create substantial technological and cognitive barriers for students, which in turn diminish their motivation.

### 2.2 Student Motivation in Software Modelling

Motivation is critical in engineering education and can be framed through, e.g., Self-Determination Theory [44], where 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 tasks itself [2, 12, 29]. 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 [11]. 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.

### 2.3 Diversity in Problem Domains and Inclusive Teaching Strategies

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 [13, 40]. Educational research increasingly underscores the importance of diversity-rich curricula, integrating multiple perspectives to build equity and inclusivity, fostering belonging and engagement [39]. 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 [10, 21, 30]. Active learning approaches, such as model construction through guided, incremental tasks, have shown promise in improving engagement and conceptual grasp in UML contexts [45]. Pérez and Rubio [38] demonstrate that project-based learning (PBL) in UML education leads to higher scores on UML examinations compared to traditional learners, suggesting that active engagement in modelling projects improves knowledge retention. PBL also enables developing transferable skills including teamwork and communication, suggesting to increase student motivation without compromising performance in other course components. This aligns with Szmurło and Śmiałek's [42] project-based framework, which shows iterative modelling tasks improve conceptual understanding compared to lecture-based approaches.

Taken together, prior work highlights persistent challenges in teaching software modelling—ranging 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.

## 3 Method

Our overall objective is to investigate how the selection of problem domains in software modelling assignments affects student motivation, engagement, and learning. We also seek to understand the extent to which these effects vary across diverse student populations. By comparing students' perceptions with those of educators, we hope to identify possible misconceptions and/or biases in the educational material, and provide recommendations for more inclusive and motivating modelling domains in software education.

### 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. 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 [25, 36].

**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 → 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 |

To ensure the effectiveness of the surveys, we first conducted a pilot study with three students and two educators. This pilot helped us evaluate the clarity and understanding of the questionnaire. The pilot participants were from three different countries and were not part in the final study. Based on the 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 the 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 provided an explanation of the study's purpose and process. We also informed them about the voluntary nature of the survey, emphasising that they could stop at any time and that all responses were anonymous. We did not require ethics approval for the study.

Table 1 shows the questionnaire for both students and educators. We provide the anonymised survey data, including questionnaires and introductory text, in our replication package [4].

*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. 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 then discussed the findings and grouped the domains into six final categories.

*Preferences (Gamification, Collaboration).* We aimed to identify the preferences of both students and educators regarding different teaching approaches, including gamification and collaboration. Based on our analysis of teaching materials and the 31 EduSymp papers, we found that gamification is often suggested as a potential approach (e.g., [9]). Additionally, it is closely linked to student motivation and engagement [43]. 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*).

*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, so we asked students whether they felt 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, 10 were undeliverable, but we found alternative email addresses for 5 of those individuals.

*Students* We invited students enrolled in courses at the authors' universities (i.e., *anonymized for review*) to fill in the student survey.

Overall, our data set 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 as implemented in R, using a significance level of $\alpha = 0.05$. We tested for differences in MO01-01 to MO01-05 and in PR02 between students and educators, student gender and student minority.

### 3.3.2 *Qualitative Analysis.*

For the open-ended responses, we adopted content analysis and thematic analysis [8] depending on the type of open-ended question. To support collaboration, we used a shared Google Sheets document where all anonymised responses are stored. Three researchers independently reviewed a subset of 20% of student and educator responses in order 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 finishing the individual coding, we triangulated a reviewing process, i.e., a second researcher reviewed the coding and in case of disagreements, the certain codes were discussed until agreement. This approach follows established practices in qualitative research that emphasise the collaborative construction of meaning rather than statistical agreement measures [8, 34].

Responses that were primarily factual or list-oriented (e.g., typical domains used) were analysed using content analysis and coded into concrete categories. According to Table 1, this includes the variables *DA02, EI02, FE01, FE02, FE03.* In contrast, responses that provided reasoning, reflections, or broader perspectives (e.g., rationales for domain selection, considerations of diversity, or strategies to encourage student engagement) were analysed 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 richness of the data.

## 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 have influenced the interpretation of responses. To reduce this risk, 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. To mitigate the risk, 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.

We provide the anonymised survey data, codebook, and introductory text in our replication package [4].

# 4 Results

The following sections present our key findings based on our quantitative and qualitative data analysis.

## 4.1 Quantitative Data

*4.1.1 Demographics.* 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 degree and 22 a master 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. 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 experience. However, we recruited only students that 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, specifically 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.

On the educator side, we recruited 22 participants, 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). One participant indicated to belong to a gender minority and two indicator belonging to another minority (without a free-text description). Finally, the educators' ages covered a broad range from 25-34 ($n = 5$), 35-44 ($n = 7$), 45-54 ($n = 4$), to 55 or older ($n = 5$).

*4.1.2 RQ2: Preferences and Design.* We asked students regarding their preferences for modelling tasks.

**Gamification** When asked about gamification (*PR02*), a clear majority stated that it increases their motivation ($n = 40$ 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's exact test, $p \approx 0.599$) as well as between students belonging to at least one minority and students belonging to no minority (Fisher's exact test, $p \approx 0.568$).

**Collaboration** Regarding collaboration practices in modelling assignments (*PR04*), showing a fairly balanced picture, 32 students stated that they prefer group projects, 29 preferred individual projects, and 21 students had no preference. There was no discernible gender difference for this questions.

*4.1.3 RQ1: Understanding Motivation Drivers.* Figure 1 depicts how motivating students find different problem domains (*MO01*). Interestingly, none of the domains shows a clear leaning in either direction. That is, the participants have differing opinions on what motivates them. In contrast, Figure 2 shows which domains educators view as motivating for students. The majority of educators believe video games and automation systems are motivating for students, while the students' answers are more polarised on this aspect. Similarly, no educator believes that video games are demotivating, and only one educator 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. Comparing students and educators, there is a statistically significant difference for their views on public information systems (Fisher's exact test, $p \approx 0.018$), automation systems (Fisher's exact test, $p \approx 0.008$) and enterprise and business systems (Fisher's exact test, $p \approx 0.004$). Similarly, for gamification, students believe that gamification increases their motivation more than educators do (Fisher's exact test, $p < 0.001$).

Comparing how motivating students of different subgroups find example domains, there is a statistically significant difference for the views men and non-men have on public information systems (Fisher's exact test, $p \approx 0.043$) and automation systems (Fisher's exact test, $p \approx 0.028$). For comparing minorities and non-minorities, there is a statistically significant difference for their views on video games (Fisher's exact test, $p \approx 0.038$).

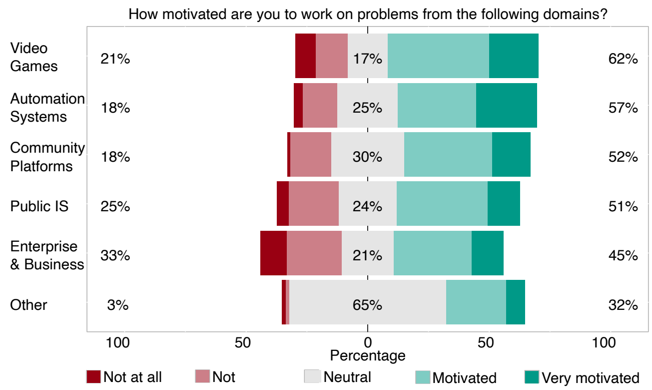


**Figure 1: Motivational Aspects of Domains Among Students.**

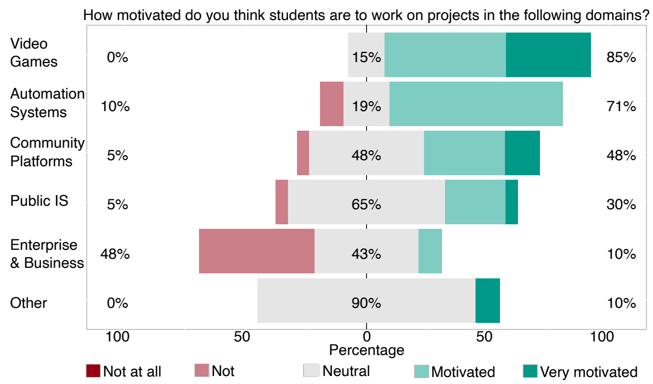


**Figure 2: Motivational Aspects of Domains Among Educators.**

## 4.2 Qualitative Data

Most courses (*DE10*) that educators typically teach are relate to core software engineering principles such as software engineering, requirements engineering, project management, software modelling and design (e.g., UML, BPMN, design patterns). Also some of the courses taught focus on programming and implementation like object-oriented programming, Java, domain-specific languages.

*4.2.1 RQ1: Understanding Motivation Drivers.* To deepen our understanding of our participants' answers regarding the motivation to work on projects related to specific domains, we asked them to provide a brief explanation for their responses to the motivation-related questions. The explanation was provided in a free text field that we then analysed qualitatively.

*Students Perspective.* The analysis of the students' answers (*MO02*) resulted in three major themes.

**Personal Interests as Primary Drivers.** The majority of students (49) answered that their motivation to work on projects related to specific domains depends on their own interests. They explained that their own interests are the main factor in deciding whether they want 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.

*Educators Perspective.* On the educators side (*MO04*) we elicited four themes of which two can be seen as main themes that were the most common answers.

**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.

*4.2.2 RQ2: Preferences and Design.* We investigated how preferences for *teaching methods* (*PR02-PR06*) and *assignment design choices* (DA01, DA02, DA04) affect motivation in modelling assignments, from the perspective of students and educators .

***Teaching methods preferences (PR02-PR06).***

**Gamification (PR03).** We asked students and educators about the role of gamification as a motivator in modelling education. *Students perspective.* Thirty-seven respondents value how gamification may *help with motivation*, and four refer to potential benefits in *increasing learning outcomes*. Thirty students report *liking gamification*. However, gamification can backfire. Five students mention how gamification can *lead to frustration*, 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.

*Educators' perspective.* Most surveyed educators see potential in *using gamification to motivate students*, but tend to take it with a grain of salt. Four educators expressed concerns about *over-reliance* *on gamification.* Three educators believe students like gamification, but two others 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

Five educators have no experience with gamification.

**Collaboration (PR05).** We asked students and educators about collaborative work in modelling assignments.

*Students' perspective.* 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).

*Educators Perspective.* In contrast to students' expressed preferences, 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 is also an argument in favour of group work. *Scalability*, both in terms of modelling challenges (3) and evaluation work by educators, also favours collaborative teams. Educators recognise the benefits of also 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.

*Educators' perspective.* Five educators make these assignments mandatory as *part of the evaluation process.* One educator suggested that a further motivation could be that the topic of the written exam 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 suggested by two educators is to highlight modelling *relevance to industry*, e.g., by drawing from their own experience. Two educators prefer highlighting the different *modelling alternatives*, while two others stress the benefits of providing *personalised feedback* to students.

***Assignment design options (DA01, DA02, DA04)***

We asked educators how they select domains, which domains they choose, and how students' interests influence this choice.

*Educators perspective.*

**Choosing modelling domains (DA01).** Student *interest*, as perceived by educators, is the most frequently mentioned criterion for selecting a domain for a modelling assignment (8). The domain's *alignment with the student profile* is considered key by five educators. One educator uses both:

> *"I mainly 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, while *fitness to the learning goals* is mentioned by four.

**Used domains (DA02).** There is a large variety of specific domains being used, 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 extent to which educators assign relevance to student's interest and motivation is a mixed bag. While 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).

#### 4.2.3 *RQ3: Inclusion and Bias in Domain Selection.*

We investigated how students perceive inclusiveness in assignment domains (*IN01–IN04*) and how educators consider diversity and bias in domain selection (*EI01–EI03*).

*Students Perspective.*

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

The majority of 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:

> *"I 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 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 three students 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.** Students highlighted four key themes (*IN03*): 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:

> *"if all assignments were to be really 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:

> *"Students in the course may come from different backgrounds. 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 inhered misogyny or phobia, in a modelling assignment it could be hard to get rid of them"* ST51

**Impact on Learning Success.** 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

Students mentioned that some domains helped them increase their motivation, such as Minecraft or sports. However, the latter was also mentioned as demotivating, thus highlighting individual preferences. In general, several mentioned relations to the real world:

> *"If the domain is close to our real 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.** 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

*Educators Perspective.*

**Considering Diversity in Domain Selection.** 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** Three themes emerged (*EI02*): educators who notice differences adjust for personal interests or background (3/16), some acknowledge potential differences but do not act on them (2), and most are unaware of differences or bias (11). For example, one observed more specific personal patterns that are related to gender:

> *"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** 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 to design inclusive experiences through a variety of domains and task choice. One educator highlighted both opportunities and challenges:

> *"[...] take a variety of domains. But for certain physical disabilities (like 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 comprehensive small examples from the student's life or experience, an inclusive medium example to train, and finally a real-world example. Note that 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 included cultural, educational, and socio-economic factors, since e.g., not everyone is familiar with how a bank loan works or

> *"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 the role of inclusivity as they thought it should be not addressed on an individual course, but on a higher level.

*4.2.4 RQ4: Feedback of Student and Educator Perspectives.* We investigated how feedback is perceived by students (*FE03*) and how educators provide it in modelling courses (*FE01*). Additionally, we asked how educators evaluate course quality (*FE02*). This comparison allows us to assess the alignment between student expectations and educator practices.

*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.** The majority of students (32) believed feedback enhances course quality, emphasising both 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.** About one third of students (19) expressed more nuanced views by linking the effectiveness to timing, lecturer responsiveness, and the visibility of integration and peer agreement. One student remarked the timing aspect since

> *"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

Several students felt their input only mattered if peers agreed or if surveys were representative:

> *"if enough people have the same opinion as me, then that [impact] can clearly be the case."* ST10

Others 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.** Almost one fifth of the participants (13) doubted that feedback influenced teaching, pointing to entrenched course structures or the 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

*Educators Perspective.* Educators described their practices (*FE01, FE02*) in three main areas: formative feedback (6), scalable mechanisms (5), and course evaluation (12).

**Formative and personalised feedback.** Educators 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 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 typically relied on standardised student surveys and university-wide processes to assess quality. Some highlighted long-term course refinement.

# 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 Key Findings

**RQ1: Understanding Motivation Drivers** Our quantitative analysis showed that educators had strong views about which domains are motivating, whereas students' opinions vary substantially. The qualitative findings confirm that both educators and students see students' interests as the primary driver of motivation. However, few educators emphasise this link to the real world, possibly due to their awareness of the inherent complexity real-world examples might introduce [12, 35].

**RQ2: Preferences and Design** Gamification can increase the motivation of most students, but it can easily backfire if ill-applied, e.g., by making it too competitive. Educators have a much more positive stance on collaboration and its benefits than students, who

**Table 2: Recommendations for software modelling education.**

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

often distrust their peers' ability or willingness to contribute adequately to those collaborations. Educators try to select interesting and suitable modelling domains to engage students, choosing domains that they expect to be familiar, or at least relatable, to most students.

**RQ3: Inclusion and Bias in Domain Selection** While most students report no exclusion, those who do point to specific issues: unfamiliar cultural references, assumptions about economic background, and limited representation of diverse identities. Students report that everyday and societal topics feel most accessible to them, while educators' awareness and strategies vary.

**RQ4: Inclusive Feedback Practices** Students value course feedback mainly when visibly acted on. Despite educators providing feedback in general, individuals might feel that their voices does not matter. This potentially affects minorities negatively.

## 5.2 Synthesis

Jointly, our findings suggest a mismatch between how educators and students perceive what makes modelling motivating. Educators have a narrower view on suitable domains or strategies such as gamification or collaboration. 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 [11, 44]. This also reframes inclusiveness: students rarely experience overt exclusion, but they do feel more included when the domain reflects their lived realities, echoing calls for diversity-rich curricula [10, 39]. Feedback shows a similar gap, 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 [12, 29].

Inclusive modelling education depends on creating assignments that make students feel represented, not just technically competent. 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.

## 5.3 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 [20]. 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" [27].

Gamification is another challenge: most students enjoy it, but some find competitive elements discouraging (*RQ2*). Educators should use a broader set of game elements to enhance engagement, ideally integrated with collaborative work. Collaboration itself divides opinion: while educators highlight its benefits, some students distrust peers or fear uneven workload distribution (*RQ2*). 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.

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 under-represented 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.

## 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.

[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*.

[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*. 103–106.

[4] Anonymous authors. 2025. https://figshare.com/s/55ebff3020d2a2dd1d7c.

[5] Jean-Christophe Bach, Antoine Beugnard, Jean-Loup Castaigne, Julien Mallet, Salvador Martínez, 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*. 123–130.

[6] Omar Badreldin, 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. CEUR Workshop Proceedings.

[7] 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*.

[8] Virginia Braun and Victoria Clarke. 2022. Conceptual and Design Thinking for Thematic Analysis. *Qualitative psychology* 9, 1 (2022), 3.

[9] 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)*. IEEE, 721–726.

[10] Sheryl Burgstahler. 2009. Universal Design of Instruction (UDI): Definition, Principles, Guidelines, and Examples. *Do-It* (2009).

[11] 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. In *Frontiers in Education*, Vol. 8. Frontiers Media SA, 1142556.

[12] 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.

[13] 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).

[14] 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*. 87–94.

[15] 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, 100–109.

[16] 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.

[17] Peter J Clarke and Alfonso Pierantonio. 2018. Teaching modeling: a software perspective. , 4 pages.

[18] 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, 206–215.

[19] Anthony J Cowling. 2005. The role of modelling in the software engineering curriculum. *Journal of Systems and Software* 75, 1-2 (2005), 41–53.

[20] 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, 120–131.

[21] 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.

[22] 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, 696–701.

[23] 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.

[24] Markus Hamann, Sebastian Götz, and Uwe Aßmann. 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*. 95–102.

[25] 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.

[26] 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*. 471–480.

[27] 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.

[28] 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, 102–109.

[29] 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.

[30] Gloria Ladson-Billings. 1995. But that's just good teaching! The case for culturally relevant pedagogy. *Theory into practice* 34, 3 (1995), 159–165.

[31] Grischa Liebel, Rogardt Heldal, 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, 18–27.

[32] Grischa 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.

[33] 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*. 107–114.

[34] 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.

[35] 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*. 1–12.

[36] 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.

[37] 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*. 309–315.

[38] Marian Petre. 2013. UML in practice. In *2013 35th international conference on software engineering (icse)*. IEEE, 722–731.

[39] 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* (2025), 1–16.

[40] 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.

[41] Martina Seidl, Marion Scholz, Christian Huemer, and Gerti Kappel. 2015. *UML@ classroom*. Springer.

[42] Robert Szmurło and Michał Śmiałek. 2006. Teaching software modeling in a simulated project environment. In *International Conference on Model Driven Engineering Languages and Systems*. Springer, 301–310.

[43] Simone Tonhão, Marcelo Shigenaga, Julio Herculani, 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*. 358–367.

[44] 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.

[45] 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.

[46] 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*. 12–21.

[47] Jon Whittle, John Hutchinson, and Mark Rouncefield. 2013. The state of practice in model-driven engineering. *IEEE software* 31, 3 (2013), 79–85.