Codesigning Ripplet: an LLM-Assisted Assessment Authoring System Grounded in a Conceptual Model of Teachers' Workflows

Yuan "Charles" Cui
yuancui2025@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Annabel Marie Goldman
annabelgoldman2025@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Jovy Zhou
jovyzhou@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Xiaolin Liu
xiaolinliu2026@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Clarissa M Shieh
clarissashieh2027@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Joshua Yao
joshuayao2026@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Mia Lillian Cheng
miacheng@gmail.com
Northwestern University
Evanston, Illinois, USA

Matthew Kay
mjskay@northwestern.edu
Northwestern University
Evanston, Illinois, USA

Fumeng Yang
fy@umd.edu
University of Maryland
College Park, Maryland, USA

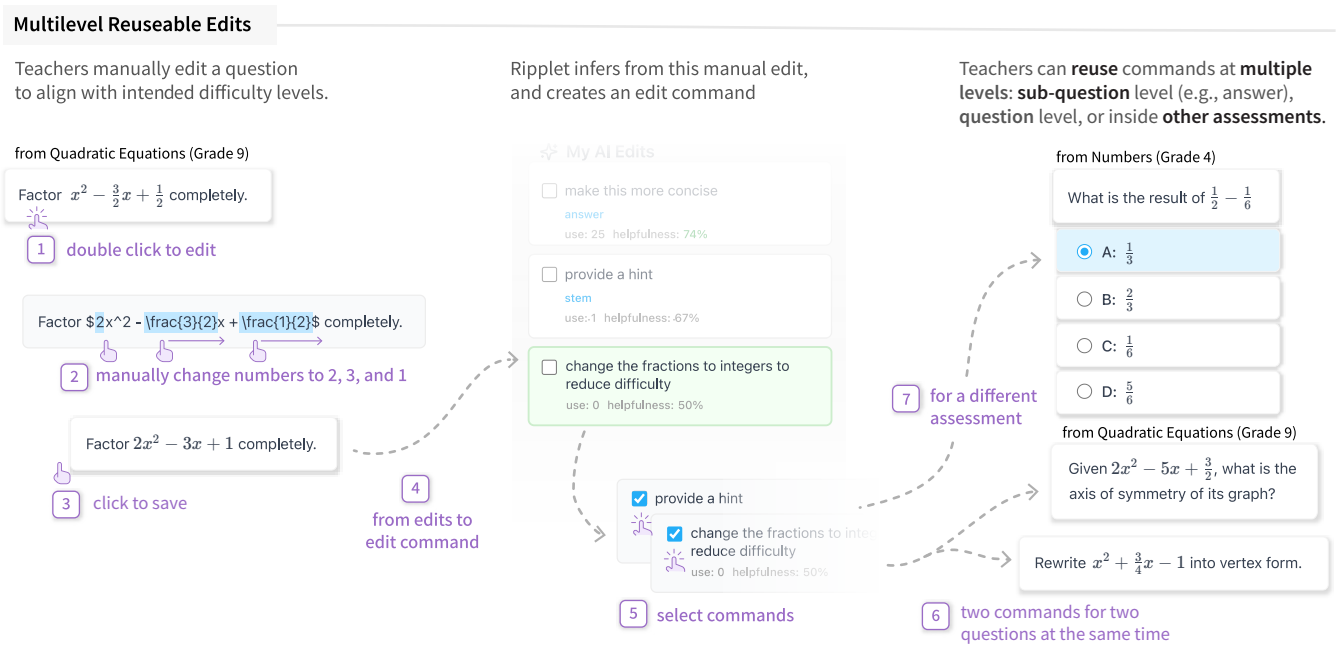

Figure 1: An example feature of Ripplet's multilevel reusable edits with LLMs: A teacher manually edits a question, and the system infers why this change was made and creates a reusable edit command, which can be reapplied to questions in other assessments.

Abstract

Assessments are critical in education, but creating them can be difficult. To address this challenge in a grounded way, we partnered with 13 teachers in a seven-month codesign process. We developed a conceptual model that characterizes the iterative dual process where teachers develop assessments while simultaneously refining requirements. To enact this model in practice, we built Ripplet,[1] a web-based tool with multilevel reusable interactions to support assessment authoring. The extended codesign revealed that Ripplet enabled teachers to create formative assessments they would not have otherwise made, shifted their practices from generation to curation, and helped them reflect more on assessment quality. In a user study with 15 additional teachers, compared to their current practices, teachers felt the results were more worth their effort and that assessment quality improved.

CHI '26, Barcelona, Spain

[1] A demo video of the system is provided in supplemental materials.

CCS Concepts

• **Computing methodologies** → *Artificial intelligence*; • **Human-centered computing** → **Human computer interaction (HCI)**; **Interaction design**; • **Applied computing** → **Education**.

Keywords

Human–AI Interaction, Education, Large Language Model, Assessment

ACM Reference Format:

Yuan "Charles" Cui, Annabel Marie Goldman, Jovy Zhou, Xiaolin Liu, Clarissa M Shieh, Joshua Yao, Mia Lillian Cheng, Matthew Kay, and Fumeng Yang. 2026. Codesigning Ripplet: an LLM-Assisted Assessment Authoring System Grounded in a Conceptual Model of Teachers' Workflows. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3772318.3790418

1 Introduction

Classroom assessments are critical in education [4, 11, 70]. Formative assessments, such as daily homework and quizzes, help students learn and teachers monitor student progress [6, 9, 11, 70, 81]. Summative assessments, such as final exams, measure student achievement and provide certification for broader audiences (e.g., parents, school administrators) [70, 81]. Together, effective assessments facilitate learning and teaching [5, 6, 11, 31] and hold all shareholders accountable in the education ecosystem [4, 10, 11].

However, developing assessments is a time- and knowledge-intensive process [42, 57, 75, 88]. Teachers need to create new questions, adapt existing materials, ensure formatting consistency, and align assessments with learning objectives [23, 32, 83]. Yet some teachers receive insufficient training or support for assessment development [58, 75]. Overworked and under-supplied teachers can produce low-quality assessments, creating inaccurate measures of students' abilities, which can potentially exacerbate disparities between high- and low-resource schools [24, 42, 75, 82].

Prior work has studied how to reduce teachers' burden of developing assessments by helping them generate individual questions. Automatic question generation (AQG) aims to fully automate question creation with minimal teacher intervention. These methods include template-based generation [91] and more recently LLM-based approaches [21, 51, 52]. While LLM-based approaches have shown promise in generating questions across diverse contexts, they often struggle to produce high-quality questions that can measure higher cognitive abilities [16, 21, 28, 30, 52]. Recent work has studied how to involve teachers in LLM-based approaches to produce high-quality questions, demonstrating the potential of human–LLM collaboration [37, 47]. However, these works mostly focus on generating individual questions and do not provide sufficient insight into teachers' current assessment authoring workflows. As teachers need to create assessments (rather than individual questions), addressing this gap is essential to build tools that can integrate into their practice.

Our goal is to understand teachers' assessment authoring workflows and build a tool to support them through human-LLM collaboration. To do so, we partnered with 13 teachers from eight U.S. high schools in a three-phase codesign process over seven months. In Phase I, we conducted formative interviews to understand teachers' assessment authoring workflows and created a conceptual model of teachers' assessment authoring practices (Section 4.1). The model describes assessment authoring as an iterative dual process in which teachers (1) create, reuse, and adapt content at multiple levels (e.g., question level, assessment level) while also (2) refining their requirements about the content of the assessment. We derived from the model a set of design objectives for helping teachers adapt content, review and structure assessments, ensure quality, and integrate requirements.

In Phase II (Section 5), we translated the conceptual model and design objectives into a system to explore how such a model could be enacted in practice. Through two rounds of design iteration, we developed Ripplet, a web-based tool for authoring assessments that embodies the process described in our model and allows us to observe how these ideas play out in real assessment authoring scenarios. Through a multilevel reusable interaction paradigm with LLMs, Ripplet supports creating questions from diverse inputs (e.g., a list of topics, PDF of a curriculum), reusing and adapting content at multiple levels (sub-question, question, assessment, cross-assessment), restructuring assessments, and inferring, tracking, and reapplying teachers' edits. Figure 1 shows one of the key features of this paradigm—multilevel reusable edits—which support iterative authoring: teachers can edit an entire question or just a part of it either manually or by prompting an LLM. The system infers reusable commands from teachers' edits. These commands can be reapplied to multiple questions or specific parts of questions in any assessment, supporting the multilevel nature of assessment authoring.

Over seven months, we collected feedback from codesign partners' independent adoption of Ripplet and observed how their usage and perceptions evolved (Section 6). Seven teachers used Ripplet in their classrooms and administered assessments to their students. We found that Ripplet was integrated into teachers' diverse workflows and helped them in their iterative dual process of authoring assessment and refining requirements. Some reported that Ripplet helped them create formative assessments they would not have had time to make. Over time, teachers' usage shifted from generation to curation with a growing sense of ownership, and they became more reflective on assessment design and quality. We also conducted a user study with 15 additional secondary school teachers, comparing their current practices (control) to using Ripplet (Section 7). Compared to control, teachers found using ripplet to be a better experience that resulted in higher-quality assessments: enjoyment (μ = +2.70 with 95% CI of [0.86, 4.54]), exploration (μ = +2.43 [0.56, 4.31]), results worth effort (μ = +1.93 [0.33, 3.53]), and assessment quality (μ = +1.11 [0.10, 2.11]) all improved on a 10-point scale.

Through this work, we contribute:

- A conceptual model of teachers' assessment authoring workflows, capturing their iterative dual process with multiple types of input and stages of action;
- A system instantiation, Ripplet, that realizes the conceptual model through multilevel reusable interactions;
- Insight from extended codesign, showing that Ripplet supports the workflows in our model and that its multilevel reusable interactions eased assessment authoring while encouraging teachers to reflect on assessment design;

- Findings from a controlled study, showing how Ripplet improves authoring experience and assessment quality.

We also discuss the limitations of our codesign partner sample and how to build sustainable and reciprocal codesign practices with teachers.

2 Related Work

2.1 Assessment Development

Developing assessments is a time-intensive process [42, 57, 75, 88], consuming up to half of teachers' professional time [76, 77]. They need to align assessments with learning objectives, devise intellectually demanding tasks, and ensure instructional relevance [23, 32, 83]. However, some teachers receive insufficient training or support for assessment development [58, 75], potentially leading to low-quality assessments and consequently uneven chances among students to demonstrate their knowledge [24, 42, 75, 82].

Researchers in education, psychology, and cognitive science have proposed measurement theories and models to guide assessment development, such as constructing detailed test blueprints [19, 86] and estimating item parameters (e.g., difficulty) [3, 29, 34]. Some argue that models of cognition and learning can inform educational assessment development, making assessments more effective in measuring student understanding [64]. However, translating such research into practice is not easy [57, 64]. In reality, teachers rarely adopt such principles when creating assessments, as these theories often lack relevance to their day-to-day instruction [57, 80]. Consequently, existing work in test development theory provides insufficient practical guidance for how teachers currently develop assessments. In our work, we first build a conceptual model of teachers' assessment authoring practices through formative interviews (Section 4) and then ground the design of our system in the conceptual model.

2.2 Automatic Question Generation

Automatic question generation (AQG) methods aim to ease the burden of creating questions [20, 42, 60], gaining momentum with the recent advances in generative AI [42, 60]. These methods range from template-based generation to neural and transformer models [1, 62, 66, 84, 85, 89, 93] and from student-authoring [59] to web-scraping techniques [15, 33]. Recent LLM-based methods dramatically lower the technical barrier to entry [17, 25, 28, 35, 41, 53, 59], and can take inputs like Bloom's Taxonomy and textbooks [21, 50–52] to generate questions of varying difficulty levels [15, 16] in multiple choice, free response, and fill-in-the-blank formats [60]. These methods typically achieve reasonable performance on quality metrics [12, 16, 45, 51, 53, 66, 89] and human ratings (e.g., answerability) [16, 25, 51, 53, 59, 74, 89].

However, there exists a gap between these automated methods and teachers' practices. Most automated solutions are not designed for teachers: some are end-to-end pipelines that require technical expertise (e.g., coding) to execute, while others use generic interfaces that lack targeted support for assessment authoring. In addition, effective assessments must closely align with individual teachers' curricula, institutional guidelines, and the unique needs of student cohorts [22, 25, 63]—requirements that cannot be met by sifting through thousands of generic questions. Assessment authoring is also a deeply personal practice that embodies teachers' pedagogical philosophy, beliefs, and years of accumulated knowledge about their students [2, 20, 46, 94]. Automated methods without careful design to involve human oversight do not fit real-world classrooms.

2.3 Human–AI Collaboration for Question Generation and Multilevel Iterative Problems

Prior work has explored ways to incorporate human oversight into LLM-based question generation [17, 26, 37, 47, 71]. TutorCraftEase allows teachers to generate questions with LLMs and accept, reject, and manually edit LLM outputs. This enables teachers to produce quality questions more efficiently [37]. Similarly, ReadingQuizMaker helps college instructors make questions for reading quizzes, providing control for when to request AI assistance; the authors found that instructors preferred this collaborative approach over an AI-only process [47]. While these systems show promise in mitigating the issues of fully automated approaches, they operate largely at the level of individual questions and provide limited support for reusing and adapting questions. Assessment authoring is inherently **multilevel**: it requires not only generating individual questions but also structuring them into a holistic assessment, tweaking specific parts of a question to fit students' needs, balancing difficulty and topic coverage, and aligning the collection to curricular goals. As we find in formative interviews (Section 4), assessment authoring is also **iterative**: teachers move between creating and adapting questions, searching for relevant materials, reviewing and restructuring the content, and refining their requirements for an assessment.

Researchers in HCI have studied multilevel problems. For example, CoLadder, a code generation system, allows users to construct a tree of prompt blocks, each block representing a different level of abstraction. This multilevel structure allows programmers to break down high-level goals into subtasks and procedural details, making the AI-generated code easier to navigate and edit across those levels [95]. In addition, DirectGPT tackles multilevel editing by enabling users to apply changes to a specific part of an object, resulting in less time spent and more preferred outcomes than standard chat interfaces [54]. HCI researchers have also designed techniques to address iterative problems with AI. PromptHive, a prompt authoring interface that supports rapid iteration on prompt variations, helps teachers create hints for educational questions by allowing prompts to be shared and reused [71]. Similarly, ABScribe, an LLM-based system for writing, supports prompt reuse by turning edits into "AI modifiers" that can be reapplied and edited [72]. Recent work by Huang et al. and Zhang et al. demonstrates how LLMs can help solve iterative problems by translating low-level UI interactions into relevant reusable macros and higher-level goals [36, 97]. Another solution to iterative problems is co-adaptive AI systems where users and systems refine their behavior in tandem [13, 27, 87]. Twin-Co, for example, improves image generation through multi-turn editing, updating output as users specify their intent [87]. We draw on these interaction techniques to design a system with multilevel reusable interactions to support assessment authoring.

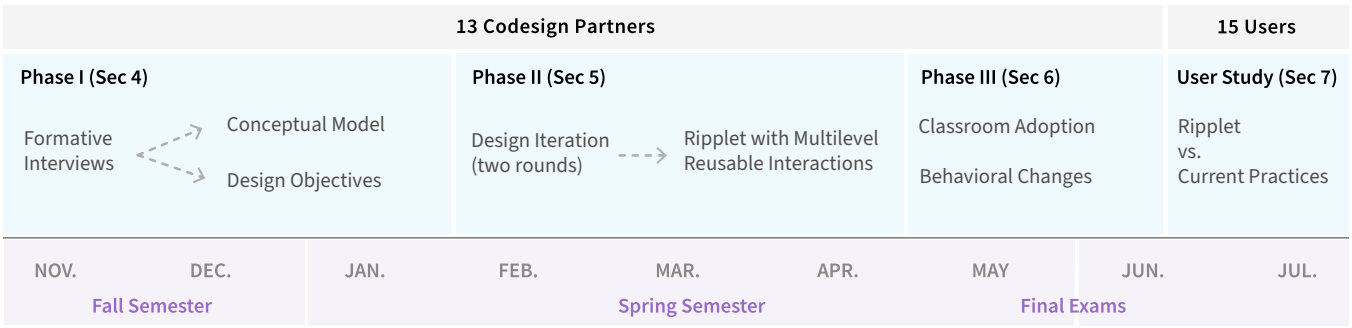

Figure 2: Overview of the three-phase codesign process and the controlled user study.

3 Codesign Overview

To design a system that integrates into teachers' practices, we need to ground our design process in their lived experiences and obtain feedback over an extended period. Therefore, we engaged in a **participatory design** process with a diverse group of teachers (Figure 2). This process consisted of **three phases** over **seven months** in a typical U.S. academic year followed by our codesign partners' schools, which comprises a fall semester (August/September – December) and a spring semester (January/February – May/June): in Phase I (Section 4), we conducted formative interviews to understand teachers' assessment authoring workflows and challenges. From there, we created a conceptual model of teachers' authoring practices and defined a set of design objectives for our system. In Phase II (Section 5), we conducted two rounds of design iteration where teachers used our prototypes and provided feedback that guided refinements toward the final system. We completed the final version before the next phase. In Phase III (Section 6), we invited teachers to independently use Ripplet to create an assessment and administer it to their students at the end of the school year. Afterwards, we conducted follow-up interviews where they reflected on their experience, their students' reactions, and the overall codesign process. After codesign, we recruited 15 additional teachers for a controlled user study (Section 7).

3.1 Participants

We began recruiting by emailing schools that have previous partnerships with our institutions. We then used snowball sampling by asking early recruits to recommend colleagues and friends. We arrived at 13 codesign partners, including 6 female and 7 male teachers from 8 schools (6 public and 2 private) across 3 U.S. states, with teaching experience from 2 to 28 years; Table 1 describes their characteristics. Everyone was 18 years or older, spoke fluent English, and was actively teaching at the time of the study. The entire codesign was approved by our Institutional Review Board (IRB),[2] and each participant received electronic gift cards as compensation (Phase I: 40 USD; Phase II: 60 USD per session; Phase III: 90 USD).[3]

[2]We withhold the IRB approval number for anonymized submission.
[3]We increased compensation in later rounds to encourage continued participation.

3.2 General Procedure

All codesign sessions were conducted one-on-one on Zoom. Before the first session, we obtained consent to record both the audio and video and provided them with the option to turn off their video. To facilitate conversations during these sessions, we used seed questions[4] that probed their practices while leaving space for them to raise questions.

Phase I: Formative Interviews. Each partner participated in an hour-long semi-structured interview to share their assessment authoring practices and challenges. We transcribed all interviews and conducted a thematic analysis [8] on all transcripts. Two authors open-coded all the transcripts. The first author then created an affinity diagram [48] based on the initial coding to extract themes. A third author then reviewed and commented on themes, and these three authors iteratively refined, split, and merged the themes through several rounds of discussion to reach a consensus [55].[5] The result is our conceptual model in Section 4.1 and design objectives in Section 4.2.

Phase II: Design Iteration. After developing the initial prototype of Ripplet, we invited our partners to participate in two rounds of design iteration sessions. Each session began with a brief tutorial—covering the entire system in the first round and new functionalities in the second—and was followed by hands-on use of Ripplet while thinking aloud. At the end of each session, teachers participated in a semi-structured interview reflecting on their experiences, likes, confusions, and suggestions for refinement. During these sessions, we took written notes on what participants did while using the system, the issues they verbalized while thinking aloud, and their responses during the interview. We also recorded the video and audio of these sessions so we could later reference them to supplement our written notes. Over three months, we continually iterated on the design of Ripplet between each codesign session using our observations of teachers' usage and their direct feedback. The result of Phase II is the final version of Ripplet (Section 5).

Phase III: Classroom Adoption. Towards the end of the academic year, we invited our partners to use Ripplet independently to create an assessment they would later administer to their students

[4]The seed questions for each phase are provided in supplemental materials.
[5]As the initial coding is not the product but the process to generate themes, we do not compute measures such as inter-rater reliability [55].

Table 1: Our Codesign Partners' Demographics and Participation. We have a diverse cohort of partners from a wide range of years of teaching experience, school types, and subjects.

Id	Sex	Years	State	School	Subjects	Formative Interview	Design Iter. #1	Design Iter. #2	Indep. Use
P1	F	22	OH	Private	Algebra, Pre-Calculus, Calculus	✓	✓	✓	✓
P2	M	26	OH	Private	History, Geography, Economics	✓	✓	✓	✓
P3	M	28	OH	Private	Biology, Ecology	✓	✓	✓	✓
P4	M	27	IL	Public	Chemistry	✓	✓	✓	
P5	M	23	IL	Public	Physics, Biology	✓		✓	
P6	F	28	IL	Public	Biology	✓	✓	✓	
P7	M	22	IL	Public	Biology	✓		✓	✓
P8	F	9	IL	Public	Biology, Health Careers	✓	✓	✓	✓
P9	M	6	OH	Public	Pre-Calculus, Calculus	✓	✓	✓	✓
P10	F	2	OH	Public	Geometry, Algebra	✓			
P11	F	6	OH	Public	Algebra, Pre-Calculus	✓		✓	
P12	M	11	OH	Public	Algebra, Computer Science	✓		✓	✓
P13	M	26	NY	Private	History	✓	✓	✓	

as a practice exam or as the official final exam. After they completed this task, we conducted follow-up interviews in which they reflected on their independent use, their students' feedback, and their overall experiences with Ripplet across the codesign process. A thematic analysis (same as Phase I) of the interviews gave us insight into how teachers used the system in authentic classroom settings (Section 6.1), and also allowed us to capture longitudinal observations on how our codesign partners interacted with the system over an extended period of time (Section 6.2).

4 Codesign Phase I: Conceptual Model and Design Objectives

We conducted formative interviews with our codesign partners. From there, we developed a conceptual model of their authoring practices and derived design objectives for our target system.

4.1 Conceptual Model for Assessment Authoring

Our conceptual model shows that assessment authoring is an iterative dual process (Figure 3) in which teachers (1) **develop the assessment** while they simultaneously (2) refine its **requirements**. In this process, teachers juggle three categories of *inputs* while moving between four major *stages* of action.

4.1.1 Inputs. We identify three categories of *inputs* to assessment authoring: (1) external rules and standards, (2) interactions with current students, and (3) available materials.

External Rules & Standards Teachers do not have complete autonomy in designing assessments; they must adhere to rules and standards from curricula, schools, districts, or states. In states like Illinois, the Next Generation Science Standards (NGSS) define what students should know and how they demonstrate understanding in K–12 science education [90]. For example, P8 created biology tests emphasizing data analysis to align with NGSS's focus on cross-cutting skills. Standardized, high-stakes exams also exert a strong influence: Advanced Placement (AP) is a program that offers university-level curricula and exams to high school students in the U.S. and Canada [7]. Teachers often mimic the AP exam formats in their tests <P1, 3, 4, 9>. As P4 noted, "*we do try to mirror the AP exam ... that exam will have some multiple choice and some free response.*" Beyond standardized content, logistical constraints such as limited exam time and quick grading requirements further restrict assessment development: "*I've gotta turn grades around within 24 hours, so I can't do a bunch of open-ended questions ... I have to write a multiple-choice test*" <P5>.

Teacher's Interactions with Current Students Teaching is a continuous interaction in which teachers observe students' progress through classroom activities and formative assessments (e.g., quizzes), then adapt assessments to suit students' specific strengths, struggles, backgrounds, and expectations. As P11 noted, "*I'm spending a lot of my job watching how my kids do the problem ... when I make the tests, I know I wanna ask the type of questions that they've been tripped up on ... I wanna see that they're getting past that.*"

Available Materials When creating assessments, teachers typically draw from various resources, including self-curated question banks, colleague-shared materials, official curricula (e.g., released past AP exams), and online sources (e.g., teacher forums). While these materials often prove useful, their quantity, quality, and relevance vary, influencing how much teachers need to adapt or create new content. As P5 noted, "*I'm not sure where they're sourcing their questions ... their diagrams are really outdated ... I'll have like an idea of what question I want to ask, and then I'll go to Google Images ... then I will just re-tailor the question to fit that diagram.*"

4.1.2 Actions. Teachers rarely define all requirements—topics to cover, difficulty levels, question formats, test length, and alignment with *students' experiences* <P1, 9, 12, 13>—at the outset. Instead, they engage in an **iterative dual** process: they (1) develop the assessment while they simultaneously (2) refine its requirements. Some begin with a vague sense of test topics, refining those topics as they explore *available materials* <P1, 2, 12>. Others draw on semi-developed requirements from past courses and focus on creating or adapting questions, making occasional adjustments to their requirements for current *students* <P8, 9>. We split this dual process into four stages: one for refining requirements, and three for developing the assessment.

Refine Assessment Requirements Teachers begin the authoring process with a set of initial ideas about their requirements, which evolve as they develop the assessment in the dual process outlined

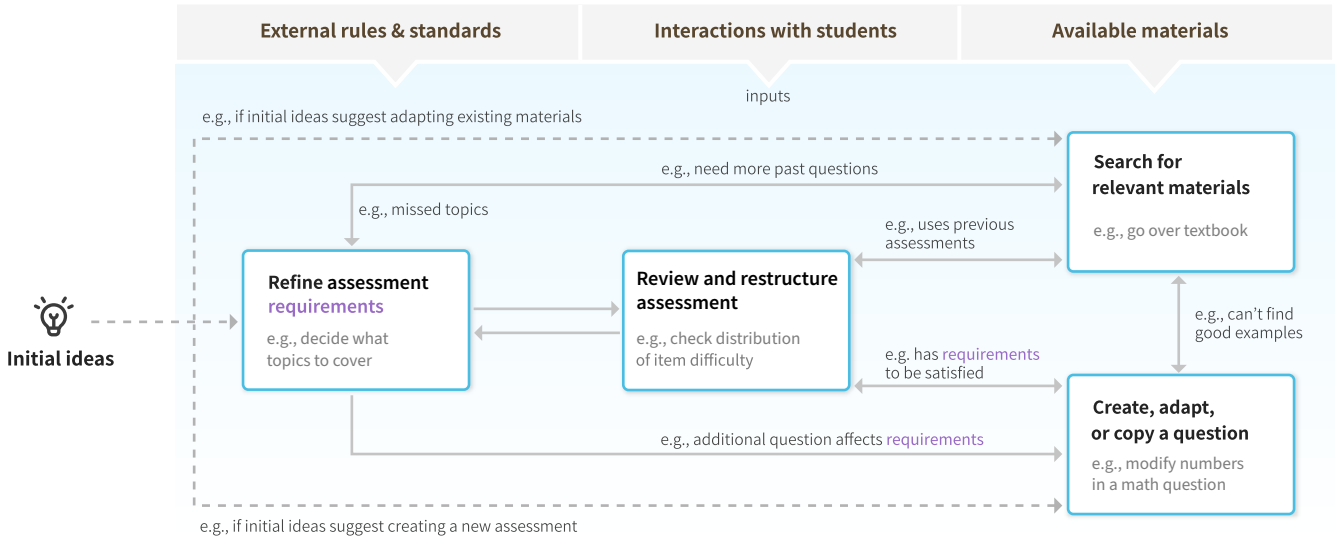

Figure 3: Our conceptual model for teachers' processes of assessment authoring. In this model, a teacher holds some initial ideas about the assessment requirements. They then start and move between the *stages* to simultaneously develop the assessment and refine assessment requirements, while considering various *inputs*.

above. Updating the requirements may prompt teachers to *search* from *available resources* or *create* new questions, and to *review and restructure* the assessment for any gaps. For instance, after reviewing past exam performance, P8 "*either change*[*s*] *the questions or* [*gives*] *students a little bit more support and scaffolding*" to improve a new test. Teachers continue refining requirements throughout the process until the final assessment meets their goals.

Search for Relevant Materials If teachers' initial ideas point them to start by adapting *available materials*, they may begin by searching for relevant content, such as course topics, past tests, homework, textbook problem sets, curriculum-prescribed question banks, or online resources: "*I can look through* [*old textbooks*] *to quickly glance and see ... that looks like it matches what I'm trying to assess from my students*" <P1>. From here, teachers may transition to *create, adapt, or copy*—adapting or copying if they find relevant materials or creating a new question if they do not. In this stage, teachers may also spot gaps that prompt them to *review and restructure the assessment* or *refine requirements*: "*I kind of look through* [*my notes and grade book*] *and say, am I forgetting anything? Is there anything that we taught that we spent time on that I'd like to test these students on?*" <P12>. Teachers often return to the search stage repeatedly throughout the authoring process to seek relevant materials.

Create, Adapt, or Copy a Question Teachers may also begin by creating new questions if their initial ideas suggest so. This is common for first-time teachers who may not have sufficient available materials to adapt <P2, 12> or those who prefer novelty and worry about test leakage[6] <P4, 13>. For instance, P13 tried to create new questions that differ from *students' expectations*: "*I try to make ... a question that presents itself a little bit differently than exactly what everybody thinks is coming.*" Teachers also adapt or copy past questions. This is often less laborious, especially for experienced teachers. If they find a helpful question by *searching* from their *available materials*, they can copy it to the work-in-progress assessment or adapt it to fit the new one; for example, P7 said: "*I've never given the same test ever ... at the very least I take a previous question and change some variables to it so they can do it in a ... different context.*" As teachers add questions to the assessment, they regularly *review and restructure* it to decide what to do next.

Review & Restructure Assessment If teachers find a past test from *searching available materials* or have a work-in-progress assessment, they may review and restructure it based on their current requirements, such as coverage of topics, adherence to *external standards*, or *characteristics of current students*: "*I tweak* [*old tests*] *every year ... I make sure that I cover the standards and the things that I've stressed within that semester*" <P9>. If teachers find the current assessment does not satisfy their requirements, they may *search* for materials or *create, adapt, or copy* questions to address the gaps: "*I pulled up the last couple years* [*of final exams*] *... we got further this year ... So the final exam had to account for a test on cell rests. So we also looked at some cell rest questions that we were gonna add to the final to make it a little beefier there*" <P7>. Teachers revisit this stage until all requirements are met, which concludes the authoring process.

4.2 Design Objectives

The conceptual model illustrates assessment authoring as a complex process. Our interviews revealed challenges in efficiently using these inputs, completing each stage, and moving fluidly between stages. For example, teachers often struggled to efficiently adapt materials across contexts, or to revise assessments without duplicating effort. Building on our conceptual model, we distill a set of

[6]Test leakage is the release or sharing of test materials that could compromise the fairness and integrity of the test; e.g., if past years' students share tests.

design objectives for an assessment authoring system.

DO1: Enable Efficient Reuse and Adaptation of Assessment Content. Teachers rarely create assessments entirely from scratch; instead, they *reuse and adapt content* from different types of *inputs* depending on their requirements—for example, using curriculum standards or targeting specific topics. As they move through the process, their requirements evolve, creating the need for efficient ways to *create and adapt questions* to match updated requirements. Some teachers may need to fill unmet requirements by creating questions on a missing topic, while others want to create similar questions to an existing one to emphasize a concept. Yet this process is often fragmented and inefficient. For example, P12 explained the repetitive burden of this scattered process: "*I'll pull up the last couple years [of exams] . . . and then try to piece them together, but it's a lot of cutting and pasting.*" Teachers often have to *search* through multiple documents, reformat content by hand, or recreate questions when small adjustments would have sufficed. A system should not only make it easy to surface content from existing sources but also help teachers adapt that material.

DO2: Facilitate Assessment Evaluation and Restructuring. Teachers emphasized that it is critical to *review and restructure* an assessment to satisfy their requirements, yet this is cumbersome in existing tools. Teachers working in Google Docs or Word reported spending a large amount of time copying and pasting questions to reorder them. Some also expressed the difficulty of randomizing options for correct answers in multiple-choice questions on a test. A system should help teachers evaluate and restructure assessments fluidly without tedious copy-pasting.

DO3: Ensure Assessment Quality. Teachers emphasized that maintaining the quality of assessments is essential but challenging. Many noted problems with existing materials: for example, P4 described how question banks often contain incorrect and low-quality questions, forcing them to solve problems himself to identify usable questions. Those who experimented with AI tools such as ChatGPT voiced similar concerns, pointing out that hallucinations and unclear phrasing made it necessary to carefully verify correctness <P3–5, 9–12>. A system should help teachers *review* and validate questions to ensure assessment quality.

DO4: Integrate Requirements into Authoring Workflow. Teachers described requirements as central to assessment authoring, yet in their current practices these requirements are rarely embedded directly into the authoring workflow. Instead, they often exist as separate artifacts—such as checklists, outlines, or mental notes—that must be manually cross-referenced during assessment authoring. Some requirements are relatively explicit and straightforward to track, such as ensuring balanced coverage of topics across a test. For instance, P5 described printing out their syllabus and marking off topics one by one to verify coverage. Other requirements are more implicit and harder to articulate directly, such as ensuring that questions are written at an appropriate reading level for students <P8>. Meeting these implicit requirements often demands significant additional effort and subjective judgment. This separation makes it easy for requirements to be overlooked and adds extra overhead to an already-demanding process. A system should provide lightweight support to ensure that an assessment satisfies a teacher's requirements.

4.3 Challenges and Opportunities for AI Support

While these design objectives are not specific to AI, recent advances in AI—particularly large language models (LLMs)—offer opportunities to help teachers achieve them. For example, prior work has shown that LLMs can generate assessment questions given text or image input, which can help teachers reuse assessment content (**DO1**). Our partners also experimented with tools like ChatGPT to draft or edit questions. However, teachers also encountered clear limitations with existing AI solutions. For instance, many found chat-based interfaces cumbersome for making targeted edits (**DO1**), often scrolling back and forth to locate specific questions or switching between multiple tools (e.g., Google Docs, ChatGPT) to adapt, label, or remove content <P1, 9, 10, 12>. The same limitation also creates a barrier for checking hallucinations and ensuring the quality of assessments (**DO3**). These challenges demonstrate that AI alone is not enough to achieve these design objectives. Instead, it requires carefully designing human–AI interactions so that AI functions are embedded in workflows that reflect real authoring practices and provide support for teachers to evaluate and refine outputs to ensure assessment quality.

5 Codesign Phase II: System Description of Ripplet

To realize the conceptual model and design objectives, we developed Ripplet, a system that leverages LLMs to support the iterative dual process of assessment authoring. In Phase II, we engaged in two rounds of design iteration with our partners and used their feedback to refine prototypes and arrive at the final system. As we refined the design, we converged upon a multilevel reusable interaction paradigm, capable of generating questions from diverse inputs, adapting content at multiple levels, restructuring artifacts, and inferring, tracking, and reapplying edits. This paradigm supports question creation (Section 5.1), basic question-level (Section 5.2) and assessment-level (Section 5.3) operations, multilevel reusable edits (Section 5.4), and cross-assessment operations (Section 5.5). We describe Ripplet's functions and their connections to the design objectives, highlight how Ripplet supports the conceptual model by marking the relevant *inputs* and *stages*, and report major refinements (🔄)[7] from design iteration. The main interface consists of four major components (Figure 4): A Menu Bar, B Search and Import Bar, C Question Cards, and D Edit Command Panel.

5.1 Question Creation

To help teachers reuse content (**DO1**), Ripplet offers five methods for creating and importing questions. Per our conceptual model, teachers often use different inputs—such as *curriculum standards*, *past exams*, or *course topics*. There are four options to add questions (Figure 4 J): teachers can (1) write questions by hand, or use LLMs to (2) generate questions from topics and (3) curriculum guides, or (4) import past assessments. Teachers can also (5) generate variations of an existing question on that question card. 🔄 This method was added after the first round of design iteration, when P1,

[7]We do not report minor refinements such as adjustments to colors or sizes. We provide interface snapshots from past versions in supplemental materials.

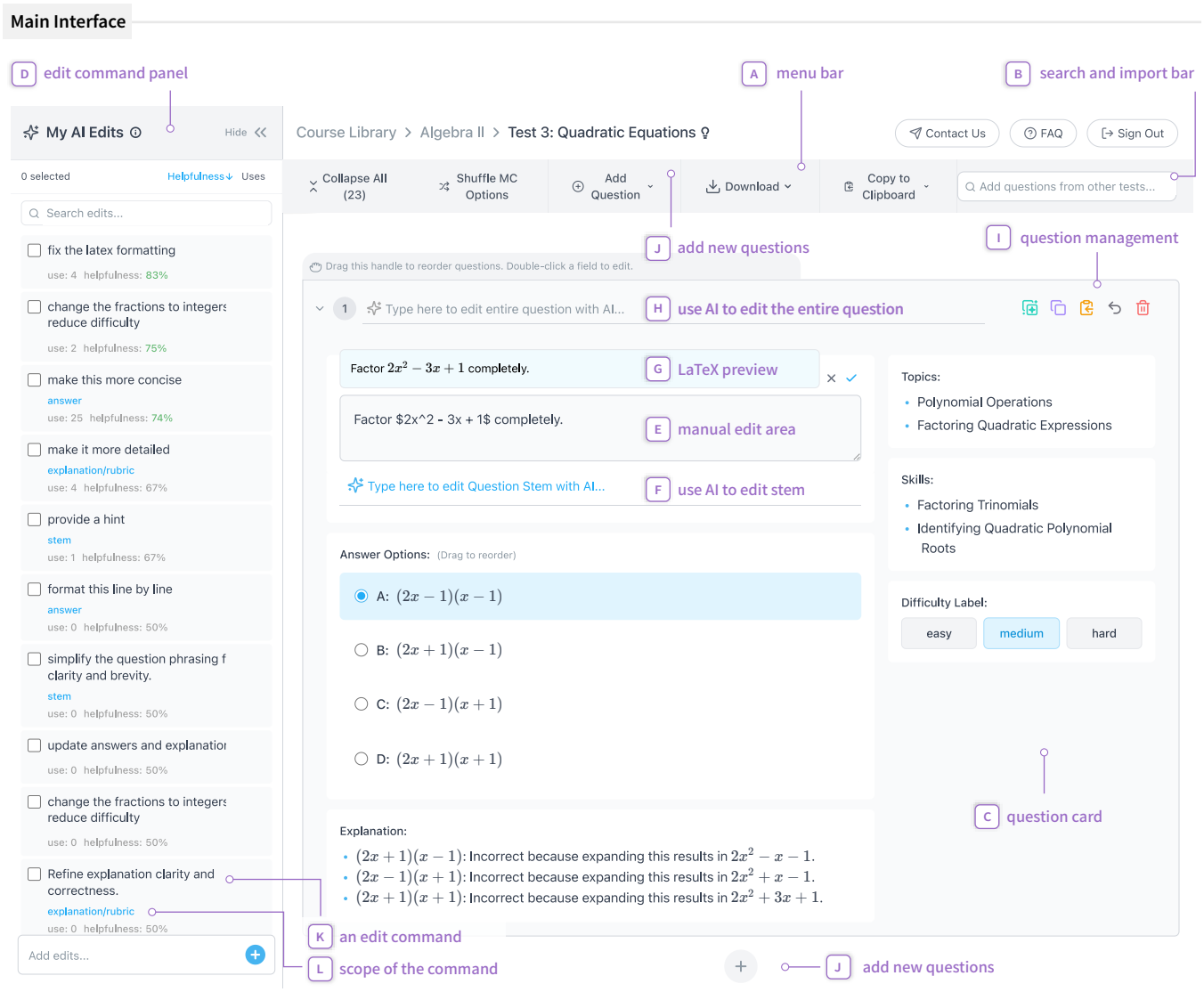

Figure 4: The main interface (after double-clicking the question stem) of Ripplet consists of four major components: A menu bar, B search and import bar, C question cards, and D edit command panel. The menu bar A allows users to add questions and navigate, restructure, and export assessments. The search and import bar B allows users to look for questions from other assessments in Ripplet and add to the current assessment. Below A and B lives the questions in the assessment. Ripplet supports two formats of questions: multiple-choice and free-response. Each question is displayed as a question card C which has six sections: question stem, answer options (multiple-choice) / answer (free-response), explanation (multiple-choice) / suggested rubric (free-response), topics, skills, and difficulty label. Users can edit a part of a question manually E or with AI F. A preview window G is shown for content that contains LaTeX. User can also edit the entire question with AI H. The five icons I allow users to manage individual question and create additional ones: using AI to generate similar questions, duplicating, copying to clipboard, undoing an edit, and deletion. Users can also add questions J. On the left side is the edit command panel D. This panel stores the edit commands K a user has made to the questions. Each command has a scope tag L that specifies the part of the question it will change once applied.

4, 6, 9 expressed the need to create variations of existing, known-high-quality questions.

5.2 Question-Level Operations

Ripplet offers a range of question-level operations that allow teachers to *edit and manage individual questions*.

Manual Edit. Double-clicking on a part (e.g., question stem) of a question triggers an inline editor for users to make direct, manual changes (Figure 4 E) from the main interface, allowing teachers to adapt questions quickly (**DO1**).

Question Management and Version Control. Ripplet provides direct actions for managing each question (Figure 4 I). Duplication and undoing changes help teachers efficiently reuse and *adapt questions* across different versions (**DO1**), while deleting allows them to remove weak or irrelevant questions to maintain quality (**DO3**). Copying a question to the clipboard allows teachers to export preferred versions and repurpose them in new contexts. These operations enable teachers to explore question variations while maintaining control and supporting iterative refinement.

5.3 Assessment-Level Operations

Beyond individual questions, Ripplet supports operations that apply to an entire assessment (), helping teachers efficiently adapt, organize, and distribute their assessments (Figure 4 A).

Restructuring. To help *restructure assessments* (**DO2**), Ripplet allows teachers to reorder questions via drag and drop. Shuffling MC options randomizes the order of options within each multiple-choice question, helping reduce answer pattern bias and quickly create alternative versions of an assessment.

Overview and Navigation. Hovering over next to the assessment name displays the composition of questions by format and difficulty, helping teachers *understand what requirements need to be satisfied* (**DO4**). Teachers can collapse or expand all question cards to move through large assessments during evaluation and restructuring (**DO2**).

Exporting. Teachers can download a PDF with or without answer keys and explanations/suggested rubrics. Teachers can also copy and paste the assessment into other software (e.g., Word or Google Docs). Expressions, equations, and special symbols are preserved as inline images to ensure that they display as expected when pasted in other software.

5.4 Multilevel Reusable Edits with LLMs

In addition to the basic operations at the question and assessment levels, Ripplet provides multilevel reusable edits which allow teachers to *adapt questions* and reuse edits at the sub-question, question, and assessment level.

Multilevel Edit. At the sub-question level, double-clicking on a specific part of a question allows teachers to prompt an LLM to edit just that part (Figure 6A; **DO1**). To show the LLM-edited version, Ripplet uses an inline difference view that highlights modifications, enabling a focused review of change and quality (**DO3**). Teachers can then accept or reject the change. This allows teachers to adapt

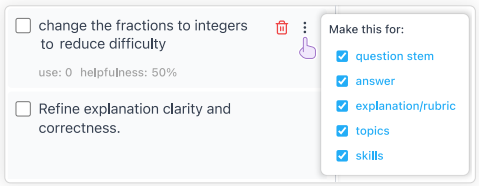

Figure 5: Each command can be tagged with specific parts of a question to ensure that only those parts are modified when the command is applied.

questions without manually editing content. This feature was implemented after the first round of design iteration, where we noticed that many partners wanted to edit only specific parts of the question to prevent unwanted changes in other parts. At the question level, teachers can edit the entire question using the top prompt bar (Figure 6B). Once the LLM completes the request, Ripplet displays a side-by-side comparison of the original and LLM-edited versions so that teachers can easily review the change and question quality (**DO3**) and decide whether to accept or reject the change. All LLM edit requests are added to the edit command panel for later reuse.

Inferred Reusable Edit. Whenever a user makes a manual edit, the system sends the previous and updated versions of the question to an LLM and requests it to infer why this change was made and generate a generalized edit command on the command panel that teachers can later apply to other questions. In Figure 1, for example, a math teacher edits a question stem to change fractions to integers. The system infers the underlying requirement (*change fractions to integers to reduce difficulty*) and adds it to the command panel (**DO4**).

Multilevel Reusable Edits. The edit command panel (Figure 4 D) tracks the edits a teacher has made and allows them to apply edits to multiple questions simultaneously, supporting efficient question adaptation (**DO1**). This panel automatically stores the multilevel edits a teacher has made and the inferred edit commands from manual edits. Teachers can also create new edit commands from scratch by typing into the text box at the bottom of the panel. Each command has two metrics: *uses* (the number of times the edit has been applied) and *helpfulness* (calculated by updating a beta-distributed prior on helpfulness with the counts of accepted and rejected edits). The panel supports sorting by either metric or keyword search to quickly locate commands.

To apply a command, teachers select it from the panel and choose one or more target questions (Figure 7). A comparison modal (right side of Figure 7) allows the teacher to toggle between questions and accept or reject the LLM-generated version individually to ensure question quality (**DO3**).

Initially, we only stored the text of each edit command without storing which part of the question a teacher was editing when that command was created. This meant that the entire question could be changed if a command was reused on another question. For example, if a teacher edited a question stem by asking to "make this shorter" then reused the resulting command from the command panel on another question, it would attempt to make every part of that question shorter. During iteration, we observed that teachers wanted to control which parts of a question to edit when applying a command from the panel. We added adjustable tags to commands

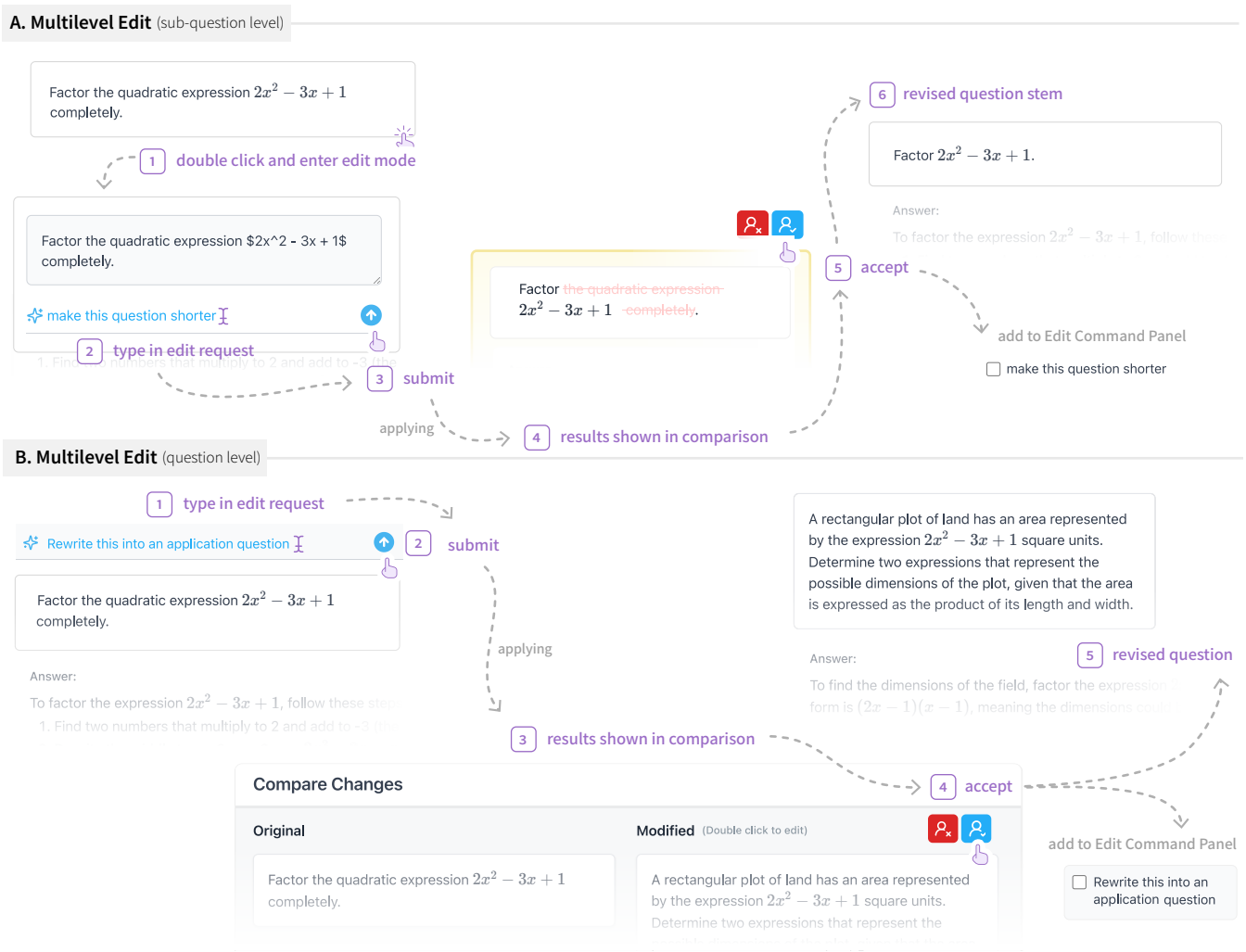

Figure 6: Multilevel edit in Ripplet. Teachers can revise either a specific part (top) or the entire question (bottom) with AI. Differences are shown inline or side-by-side for quick review.

that specify which parts of a question it should apply to (Figure 5).

Reusable Commands As a Way to Enact Requirements. The reusable commands not only help teachers adapt assessments, but also better *integrate teachers' implicit and explicit requirements* into assessment authoring by tracking and surfacing their edits (**DO4**). Initially, we experimented with designs where requirements are reified in the main interface, e.g. as a nested checklist. We imagined the system could help teachers write down and mark off items on that list by identifying which question(s) contribute to each requirement. We quickly realized a reified requirement checklist does not fit the implicit and entangled nature of assessment requirements. Some requirements are difficult to articulate up front, such as using only positive numbers in a question about factoring so that students can focus specifically on factoring. Other requirements are entangled: for example, when teachers adjust question difficulty, they might simultaneously change the coverage and progression of topics in the assessment. It would be difficult to isolate and articulate all such requirements explicitly. However, we noticed that teachers often discover requirements as they work on specific questions. We designed multilevel reusable edits to capitalize on this behavior by eliciting, tracking, and surfacing a teacher's requirements implicitly from their edits, without imposing the burden of writing down explicit requirements.

Managing the Command Panel with Similarity Checker. Teachers often make many edits to an assessment. During design iteration, this could generate many edit commands and quickly overwhelm the command panel, reducing its usefulness. While teachers could use the search and delete functions to manage the panel, this requires additional effort. However, many edits are similar. For example, P3 repeatedly removed extra words to keep question stems concise, and P12 repeatedly used AI edits like "add a hint" or "make the question easier by giving additional information". To reduce the length of the command panel, we implemented an LLM-based similarity checker. When a new command is generated, it is passed to an LLM along with the current list of commands on the panel to judge if the new command is sufficiently different from all existing

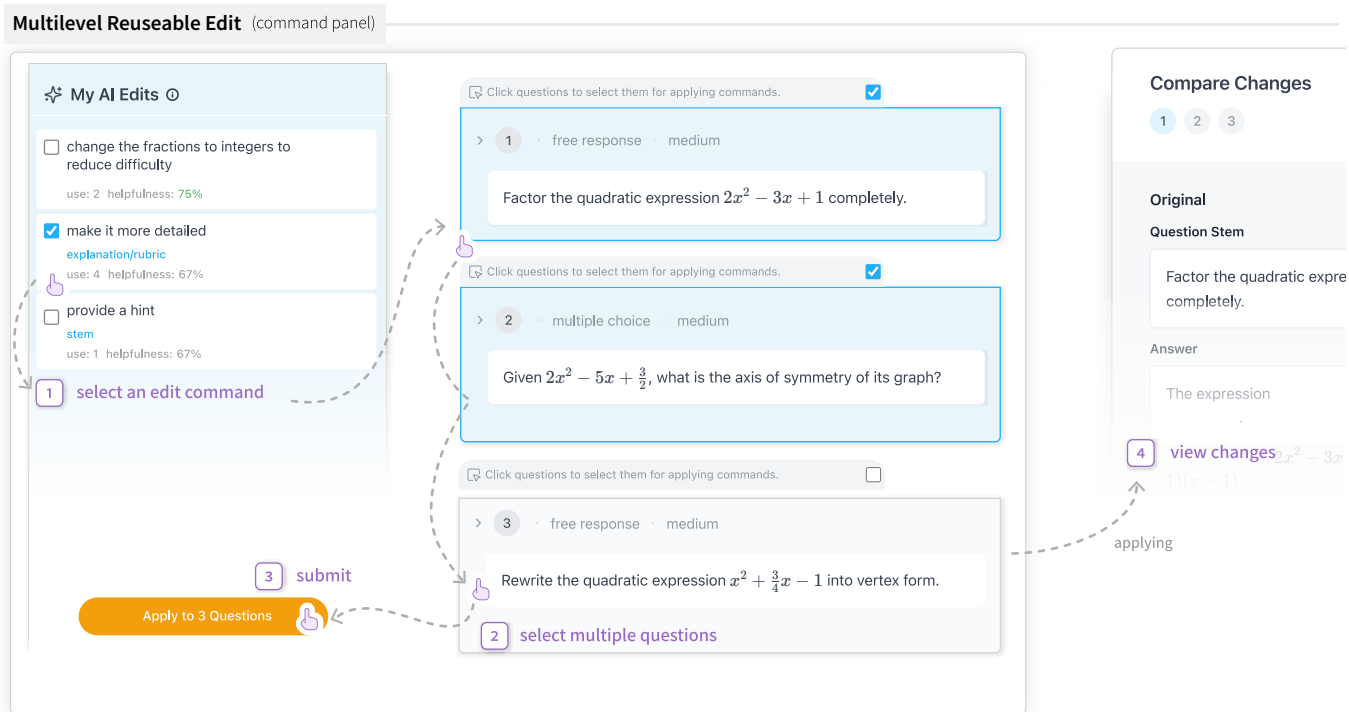

Figure 7: Multilevel reusable edits from the command panel. Teachers can select a command and apply it to specific questions. They can review the AI-generated changes with the original question side by side and accept or reject each change individually.

commands. To ensure consistency, we collected common examples of similar commands and used few-shot prompting for the checker's prompt template to help define command similarity (e.g., "make it shorter" and "make more concise" are similar). The similarity checker significantly reduced the number of repeated commands on the panel and made the panel more usable for teachers.

5.5 Cross-Assessment Operations and Supports

Ripplet also provides tools that operate across assessments to help teachers reuse and adapt materials (**DO1**).

Sharing Edit Commands across Assessments. Initially, the edit commands were only preserved within an assessment. Multiple partners, including P1 and P12 who used Ripplet extensively, indicated that they needed to apply similar edit commands across assessments and even across courses. Therefore, we made each teacher's edit commands available across their assessments to help them adapt other assessments without re-specifying commands (**DO1**).

Importing Questions from Other Assessments. To support reusing high-quality content (**DO1**), Ripplet allows teachers to add questions from previous assessments in the system via the search and import bar (Figure 4 B). Teachers can *search and filter past assessment questions* by a variety of criteria, such as class, topic, format, and difficulty, then view full question details and directly insert questions into the current assessment.

Equation Authoring and Syntax Highlighting. In our formative interviews, mathematics and science teachers described significant challenges in creating, editing, and formatting expressions and equations using existing tools like Microsoft Word. For example, P12 struggled to create a cube-root symbol, so they used the square-root symbol available in their software and hand-wrote the "3" on every copy of the test. To support a wide range of symbols and display them properly, Ripplet enables users to render and author expressions and equations using LaTeX notation. Recognizing that teachers typically lack familiarity with LaTeX,Ripplet allows teachers to generate and edit LaTeX symbols with an LLM and automatically displays a rendered preview whenever mathematical expressions or markdown blocks are detected.

To support computer science teachers, during the second round of design iteration, we implemented syntax highlighting for all major programming languages using the same tab interface for displaying rendered results (`Answer: for i in range(1, 6): ...`).

5.6 Ripplet Walk-through

To demonstrate how Ripplet supports assessment authoring and realize the conceptual model, we walk through a fictional scenario where a teacher, Ms. Ripley, creates an assessment (Figure 8). We highlight the connections to the conceptual model by marking the *inputs* and *stages*.

Ms. Ripley is teaching an Algebra II course that has a test coming up on linear equations. She logs in to Ripplet, enters her course

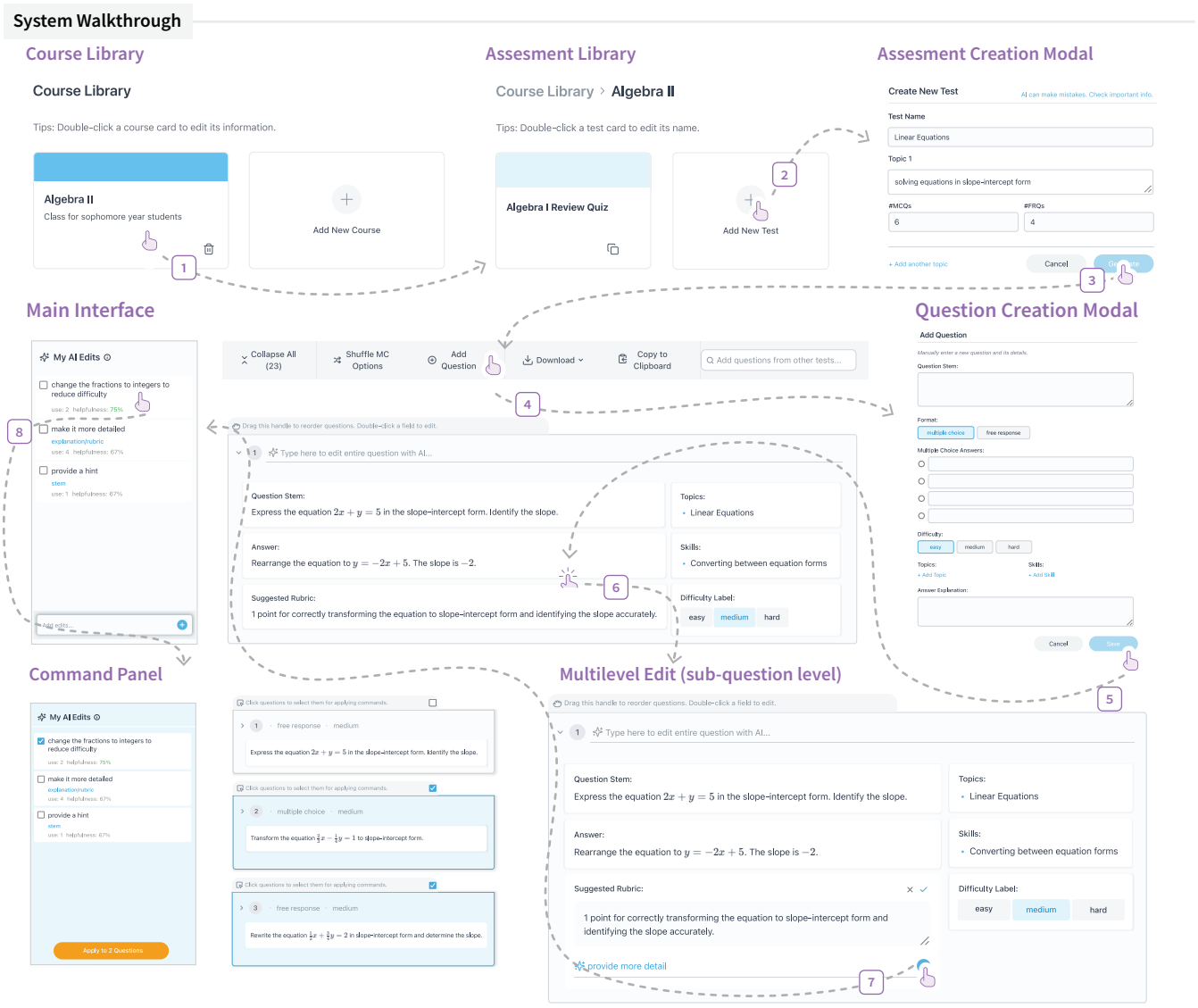

Figure 8: System walkthrough of Ripplet. Ms. Ripley starts assessment creation by generating from a topic. She then manually writes a question and uses AI to edit it. Finally, she notices that some questions are too hard, so she applies an AI edit command to edit multiple questions at once. She reviews the revisions and accepts the suitable changes to finalize the test.

library and clicks on her Algebra II course. She enters the assessment library and begins creating her test. Ms. Ripley sees several ways to do this and chooses "Generate from Topic" because she has specific topics in mind: *solving equations in slope-intercept form* and *converting them to standard form*. She also has a rough idea of how many multiple-choice and free-response questions a balanced test should have to take up the full class period.

After she clicks "Generate", Ms. Ripley enters the main interface with the assessment. After quickly *reviewing* the questions, she decides to *add additional questions* to *cover a missing topic*. This opens a modal with the same question creation methods she saw before. This time she *writes a new question* on linear equations from scratch. Now that her test satisfies her length and topic coverage requirements, Ms. Ripley carefully *reviews* and *revises each question*. She notices that the suggested rubric for a free-response question is not detailed enough, so she double clicks on it and asks AI to provide more detail. She *checks the AI's output* and notices that the new rubric is too detailed, so she rejects the change and edits it manually. She then notices that some questions may be *too hard for her students* because the solutions involve fractions, which her students have been struggling with. For this test, she only wants to assess students' ability to work with equations of a line, not their understanding of fractions, so she reuses the command "change the fractions to integers to reduce difficulty" in the command panel and applies it to all corresponding questions at once. She reviews the changes and accepts the suitable ones. Once satisfied, she downloads it as a PDF to hand out in class.

5.7 System Implementation

We use OpenAI's GPT-4o for Ripplet's LLM-related features and specifically GPT-4o with vision for generating questions from curriculum guides and importing questions, as it can extract special symbols and expressions from images of documents. Some prompt templates benefited from collecting examples from codesign partners. For example, we used common examples of similar commands to revise the prompt template of the similarity checker.

In the database [8], we store the full edit history of questions for version control. We also store the full history of edit command usage to calculate the helpfulness and use metrics. In the interface, we render mathematical equations for in-browser display and copy-to-clipboard using `remark-math` [69] and `rehype-katex` [68], while for PDF rendering we use `KaTeX` [38]. Similarly, for syntax highlighting, we use `react-syntax-highlighter` [67] for in-browser display and copy, while `prismjs` [65] is used for rendering PDFs.

6 Codesign Phase III: Independent Use and Longitudinal Observations

Our codesign process provided a unique opportunity to track Ripplet's use across two complementary dimensions: independent adoption and longitudinal engagement. Seven teachers used Ripplet independently to create and administer assessments in their classrooms, giving us insight into how well Ripplet functioned in organic settings and captured the the workflows in our conceptual model. In addition, because many of the same teachers participated in multiple rounds of codesign sessions and continued corresponding between sessions, we were able to observe how their practices and preferences evolved over several months.

6.1 Independent Classroom Adoption

Following the second round of design iteration, seven teachers used Ripplet to create assessments and administered them to their students. They created a variety of assessments—from official final exams for state-required classes to practice tests for dual-credit community college courses.

Ripplet Integrated into Teachers' Existing Workflows. Teachers' independent use revealed that Ripplet fits naturally into a variety of existing workflows by supporting different *inputs* and *stages* in the conceptual model. For example, P9 *found high-quality AP questions*, imported them, and used the "Generate Similar" and AI editing features to create a review packet, while P8 uploaded *curriculum standards* to generate a mock exam for students in their health careers course and used a mix of manual editing and the command panel to *reduce the reading level* of some questions. P1 started with topic-based generation to target specific areas; some questions *inspired and reminded them to cover a different topic* and blend multiple topics together into one question, so they created new ones both manually and with AI. Finally, they *reordered the questions so that they progress in difficulty*. By supporting diverse question generation, adapting questions at multiple levels, and restructuring assessments, Ripplet allowed teachers to build on their materials and navigate the iterative dual authoring process in the conceptual model without abandoning their established ways of working.

[8] We provide the database design documentation in supplemental materials.

Time Savings Enabled Teachers to Create Materials They Could Not Before. Teachers expressed that Ripplet enabled them to create assessments they previously lacked the time or resources to produce. P8, for instance, created a 28-question practice final exam in just 30 minutes by using curriculum standards and applying reusable edit commands to *scaffold questions and adjust language level*—a task that would normally have taken them over five hours. They emphasized that the efficiency made the creation of more formative assessments possible: "*I just wouldn't have made it otherwise.*" Similarly, P9 explained that without Ripplet, they would have had to purchase a commercial review packet, but instead they were able to build one themselves quickly and give it to their students. Teachers noted that these new assessments had clear downstream benefits for students: P8's class treated the mock exam as seriously as their actual final and learned a lot by working through the mock exam, and P9's students reported that the review packet was helpful in preparing for the final exam. These accounts underscore Ripplet's educational impact of not only saving teachers' time but also expanding the opportunities they could offer their students.

Multilevel Organization Eased Assessment Refinement. Teachers also praised how Ripplet's structured interface supported the stages of assessment authoring, particularly restructuring assessments and revising questions. The system's card-based question display and menu bar helped them *reorder questions* and *shuffle multiple-choice options*—tasks that were cumbersome in tools like Google Docs, which required manually cutting and pasting text. P12 described the organizational structure as "*the huge benefit with Ripplet ... each question is its own element, and it's easy to make localized changes.*" Being able to edit specific parts of questions with AI further supported this process by allowing teachers to *refine multiple-choice distractors* or *rephrase question stems* without disturbing the other parts of questions <P3>. The multilevel organization gives teachers more control over how their assessments were assembled, supporting evaluation and restructuring while reducing the tedium of manual formatting.

6.2 Behavioral Changes in Assessment Authoring

Evolving Ownership: from Generation to Curation. As teachers engaged with Ripplet over time, their sense of ownership and role in assessment design shifted. Early in the codesign process, our partners spent more time generating questions with LLM-based methods. While they emphasized the need to reuse and adapt prior materials, they did not utilize many features designed for that purpose (e.g., importing assessments, command panel). As a result, many perceived AI as the "real" author and themselves as proofreaders, often questioning whether they could truly claim ownership of the assessments. With repeated use, however, their behavior and perception evolved. Teachers routinely imported questions from their own past tests, modified them for new contexts, and wove them into new assessments. P12 highlighted the efficiency of being able to "*add questions from other tests and filter them.*" Similarly, the command panel for reusing edit commands—initially underexplored—grew into a valued tool. In the first round, some overlooked the command panel entirely; by the second, they praised it as a timesaver that improved productivity <P4, 8>. Teachers also grew

comfortable deleting low-quality questions, rejecting weak AI suggestions, and refining assessments manually. Teachers increasingly recognized themselves as the primary creators. P9 explained that at first they felt "*the AI did most of the work,*" but later came to see that they were "*the one deciding what to keep, what to change, and what made sense for my students.*" In addition to teachers' growing familiarity with AI and their development of more effective prompting strategies, several features of the system likely contributed to this transition, such as the side-by-side comparison modal and inline difference view of AI editing, which allows teachers to control the quality (**DO3**) of assessments by reviewing, accepting, or rejecting AI edits easily.

Becoming More Reflective by Writing and Seeing Commands. Teachers also reported that Ripplet made them think more deliberately about assessment design and question quality. P1 explained that using AI to adapt questions forces them to think more critically about what it is that makes a question difficult. Likewise, P3 reflected that their usage of Ripplet encouraged them to slow down and become "*more deliberate in what I do, instead of just going through the motions.*" Similarly, P12 expressed that seeing the commands showing up on the command panel helped them *understand their own requirements* and reflect on what needed to be done to design a high-quality assessment (**DO4**). By inferring and surfacing teachers' requirements as reusable edit commands, the system helped teachers approach assessment authoring in a more systematic and reflective way.

Extending Ripplet's Applications in Teaching. Extended engagement also reshaped how teachers envisioned their professional practice. P8 described wanting to use Ripplet on demand during office hours to generate practice questions, while others imagined using it for multilingual or remedial contexts (Ripplet supports the display of text in multiple languages). Teachers also highlighted time saved outside of assessment authoring: P3 and others noted that Ripplet's answer keys and explanations allowed them to hand these directly to students, reducing the need for lengthy office hours. P1 explained that students can self-diagnose first instead of them hosting a multi-hour session to explain the answers. These reflections show that teachers came to see Ripplet not only as an assessment authoring tool but also as part of a broader ecosystem of teaching, learning, and collaboration.

7 Controlled User Study

While our codesign process showed how Ripplet supported diverse workflows in the conceptual model and how teachers engaged with it over time, it involved codesign partners who were deeply familiar with the system. To evaluate Ripplet on a sample of teachers less familiar with the system or invested in its design, we conducted a controlled, within-subjects study with 15 other teachers. Each teacher created assessments in two conditions: once with their current practice (control) and once with Ripplet. This design allows us to make a direct comparison between Ripplet and their current practices. It can also provide evidence on whether Ripplet's benefits and integration to teachers' workflows generalize beyond our codesign partners.

Table 2: Information about teachers in controlled user study.

Id	Sex	Years	State	School	Subjects
U1	F	19	OH	Private	Geometry, Pre-Algebra
U2	M	8	OH	Public	Chemistry
U3	F	18	VA	Private	Economics, Human Geography, Business
U4	F	30	OH	Public	Biology, Anatomy
U5	F	1	IL	Public	Biology, Physiology
U6	F	8	TX	Public	Biology
U7	M	31	OH	Public	Middle School History
U8	M	13	TX	Public	Economics, Financial Literacy
U9	F	27	OH	Private	History, Psychology
U10	F	7	OH	Public	Pre-Algebra, Algebra I, Geometry
U11	F	5	OH	Public	Algebra I
U12	F	6	OH	Public	Algebra II, Statistics
U13	F	20	OH	Private	Middle School Math
U14	F	29	OH	Public	Financial Literacy
U15	M	15	IL	Public	Computer Science

7.1 Participants

We asked our codesign partners to reach out to their teacher networks through mailing lists and Facebook groups. Table 2 shows the 15 teachers (11F, 4M) we recruited from 13 schools in 4 states, using the same criteria as codesign (see Section 3.1). The group included 13 high school and 2 middle school teachers across different subject areas: mathematics and computer science (6), natural sciences (4), and social studies (5). Their teaching experience ranged from 1 to 31 years. All participants completed a pre-survey that included questions about their experience and comfort with using AI tools. Some had never used AI at work and self-identified as having "no ability" in using AI <U14>, while others identified as "advanced" users and had used tools such as ChatGPT to create and adapt assessments frequently <U2>. The pre-survey also asks each teacher to list the courses they will teach next semester and two chapters or units in each course that are similar in length and difficulty. None of the participants had previously used Ripplet or seen its interface; their only prior exposure was the recruitment message indicating that Ripplet is an AI-powered tool for authoring assessments. This study was IRB-approved, and each participant received a total compensation of 195 USD.

7.2 Procedure

The study involved two Zoom meetings, each followed by an asynchronous task that teachers conducted and recorded themselves. To counterbalance potential order effects, we randomly assigned participants to either a control-first or Ripplet-first group, then randomly picked a course they listed in the pre-survey and randomly assigned two units from that course to the two conditions.

In the control-first group, participants began with a 15-minute Zoom session where we introduced the study and outlined the first asynchronous task. In this task, we asked teachers to create an assessment for a unit of a course that they would teach the next

[9]Although the 1–10 scale does not have a neutral midpoint, its granularity allows sufficient gradation in responses, and because our analysis focuses on differences across conditions, the absence of an explicit neutral point is unlikely to affect the results.

Table 3: Survey items grouped by five areas. Participants were asked to rate each item on a scale from 1 to 10.[9]

Area	**Survey Item**: (highly disagree [1] - highly agree [10])
Enjoyment	I would be happy to use this way of creating assessments on a regular basis. I enjoyed using this way of creating assessments.
Exploration	It was easy for me to explore many different ideas, options, designs, or outcomes, using this way of creating assessments. This way of creating assessments was helpful in allowing me to track different ideas, outcomes, or possibilities.
Results Worth Effort	I was satisfied with what I got out of this way of creating assessments. What I was able to produce was worth the effort I had to exert to produce it.
Perceived Control	I felt I had a say in the assessment creation process. I was able to influence the assessment creation process.
Assessment Quality	The assessment I created can measure my students' skills and knowledge or help them improve. The assessment questions are worded clearly. The assessment is adequately difficult for my students.

semester, using their current method of authoring assessments. While working on the task, they recorded their screens and narrated their process by thinking aloud. Afterward, they completed a survey (Section 7.3) reflecting on their experience. The second Zoom meeting (60 minutes) began with a 10-minute tutorial video introducing the Ripplet tool followed by 40 minutes of hands-on exploration. During this time, participants experimented with Ripplet and received support from the research team as needed. In the last 10 minutes, we introduced the second asynchronous task: creating an assessment using Ripplet for the unit assigned to the Ripplet condition. As with the first task, the participants recorded their screens, narrated their process, and completed a post-task survey. In the Ripplet-first group, participants followed the same set of activities and tasks, but in reverse order.

7.3 Survey Design

We asked participants to evaluate Ripplet and their current practices in five areas through an 11-item survey (see Table 3). As assessment authoring is a open-ended creative task, we first adapted the Creativity Support Index (CSI), which offers a meaningful and validated way to assess how well a tool supports creative work [18]. From CSI, we selected the areas applicable to evaluating our system: "Enjoyment", "Exploration", and "Results Worth Effort". Because we need to administer the survey in both the control and Ripplet conditions, we rephrased the CSI statements that read "the/this system or tool" to "this way of creating assessments". To understand users' sense of agency and control in a system involving LLMs, we measured perceived control with two questions from Lee et al. [43] and adapted them by changing "content editing" to "assessment creation" to suit our setting. Finally, to understand the educational impact of Ripplet, we added three questions regarding the quality of assessments. We adapted two questions from Cui et al. [21] for measuring the relevance and clarity of assessments, and we added another question on the difficulty of the assessments. We asked teachers to self-evaluate assessment quality rather than rely on external raters, because what constitutes a "high-quality" assessment differs across school settings and student cohorts even within the same course, making it difficult to define a universal, objective standard [56, 79]. We also invited five codesign partners (P1, 3, 7, 12, 13) for feedback on the survey. P7 expressed that only those who know the students' abilities and struggles can fairly judge assessment quality, supporting our decision to use self-evaluation. We finalized the survey after codesign partners confirmed that the questions capture the most important aspects of assessment quality.

7.4 Quantitative Comparison

To compare participants' authoring experiences and the quality of assessments from using Ripplet vs. their current practices (control), we compared their ratings in each of the five areas on the survey.

Analysis. First, based on the guidance for analyzing Likert items, we averaged ratings for items within each area for each participant and condition [14, 61]. For each area, we then compared Ripplet vs. control using paired-samples t-tests on the within-participant mean differences (Ripplet − control). To account for the five parallel tests, we controlled the false discovery rate using the Benjamini–Hochberg (BH) procedure ($\alpha = 0.05$), and we report the BH-adjusted p value, mean difference, and its 95% CI for each of the five areas. We chose the paired t-test because it is high-power and sufficiently robust to modest assumption violations [61, 78, 92].[10] This analysis plan was pre-registered on OSF.[11]

Results. Figure 9 shows the distributions of participants' ratings for the five areas in both conditions, as well as the mean and the confidence interval of the difference in ratings (Ripplet − control). **Ripplet shows significant improvement over control in four areas**: enjoyment ($\mu = +2.70$ with 95% CI of $[0.86, 4.54]$, $p = 0.003$), exploration ($\mu = +2.43$ $[0.56, 4.31]$, $p = 0.012$), results worth effort ($\mu = +1.93$ $[0.33, 3.53]$, $p = 0.012$), and assessment quality ($\mu = +1.11$ $[0.10, 2.11]$, $p = 0.032$). Ratings for perceived control are similarly high across conditions, possibly because teachers in both settings could manually create and edit questions.

7.5 Qualitative Findings

In addition to rating both conditions, participants answered two open-ended questions: (1) which features or functionalities they found most helpful, and (2) the difficulties they encountered. We also coded[12] participants' self-recorded usage of Ripplet to understand their workflows and features used. Their responses and the

[10] t-tests can be performed on low-number-of-item Likert scales because parametric statistics are robust with respect to Likert being ordinal [61].
[11] OSF link: https://osf.io/j8zey
[12] We coded the videos by indicating the specific features users used in Ripplet. The codes are provided in supplemental materials.

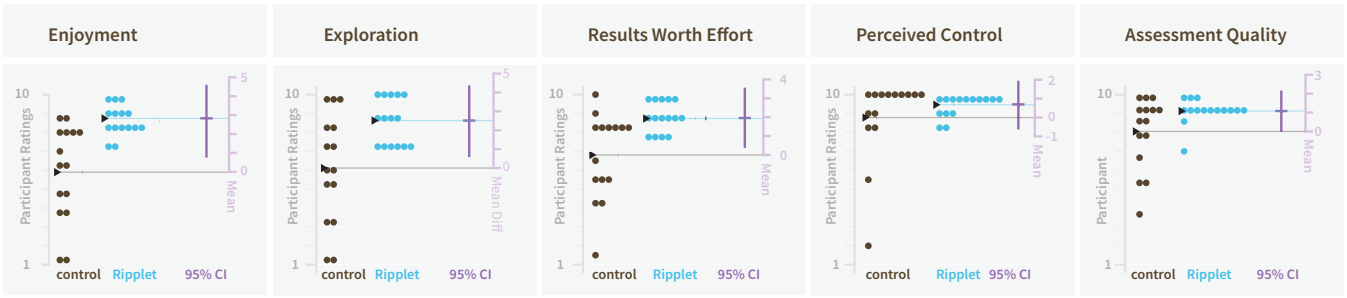

Figure 9: Distribution of teachers' ratings for each of the five areas in both conditions, with mean differences (Ripplet − control) and their 95% CIs from the paired t-tests.

patterns we observed in their self-recordings corroborate our observations from the codesign process (Section 6). Many users began by importing *curriculum standards* or *past assessments* then used a mix of manual edits and AI edits to *adapt questions* and *reorganize assessments*. Participants highlighted that multilevel reusable edits were efficient for revising assessments <U3, 5, 8, 10, 12>. For example, U8 applied the command "reword this question so that a non-native speaker could understand it" across several questions to accommodate students who needed language support. Teachers also valued features such as generating similar questions, which allowed them to reuse high-quality materials <U2, 13>, and shuffling MC options <U9> to *restructure assessments*. U8 noted that Ripplet would enable them to create more formative assessments for the upcoming school year—assessments they otherwise would not have had time to prepare.

At the same time, several math teachers mentioned their unfamiliarity with LaTeX and often had to ask AI to correct related issues <U11-13>. U13 expressed confusion that similar prompts sometimes produced different results, suggesting that understanding the uncertain nature of LLMs and adjusting expectations could take time. Teachers also expressed a desire for support in generating questions with visual representations <U5, 8>, suggesting an important area for future work on assessment authoring systems.

8 Discussion, Limitations, and Future Directions

We discuss limitations of our work and directions for future research. We reflect on how characteristics of our teacher cohort shaped the conceptual model and system use; we discuss how uneven risks of AI reliance affect different teachers, motivating the need for safeguards and supports tailored to these differences. Finally, we propose deeper evaluations for multilevel reusable interactions and reflect on the lessons learned from sustaining a long-term codesign process with teachers.

8.1 Variations in Teacher Backgrounds Affect Ripplet Usage and Generalizability of Conceptual Model

Because our codesign partners were largely STEM teachers and experienced teachers (Section 3.1), many of the behaviors, challenges, and workflows that informed our conceptual model and shaped our system design could reflect practices more common in math and science assessment authoring as well as the habits of veteran teachers with well-developed materials. Prior work shows that STEM and humanities teachers often have different instructional priorities and needs [73], a distinction that also surfaced in our sample: STEM teachers often invested much effort in verifying the accuracy of the answer keys and tweaking numerical details, making *reviewing assessments* and *adapting questions* critical stages in the conceptual model. On the other hand, the two social studies teachers (P2, 13) created more free-response questions and did not spend significant time scrutinizing the answers, shifting the burden of verifying answers to grading rather than authoring. New teachers in their first or second year and those from humanities and social studies were underrepresented, limiting our ability to observe how their distinct disciplinary practices and early-career needs might surface different inputs, stages, or transitions within the conceptual model. As a result, the generalizability of our conceptual model may be constrained by the makeup of our codesign sample. Future work could expand the model by recruiting a more balanced set of teachers across subject areas and years of experience to support the full diversity of K–12 teaching contexts.

8.2 Mitigating Uneven Risks of AI Reliance Needs Carefully Designed Safeguards

Given teachers' concerns around fairness and safety in AI-generated content [96], it is important to consider how variations in teacher backgrounds may produce varied risks. Teachers who are already stretched for time and materials may be particularly vulnerable to over relying on AI outputs without the capacity to carefully review them. In contrast, teachers with more experience, training, or institutional support may be better positioned to reuse high-quality content, recognize hallucinations, and adjust their use accordingly. Safeguards and training should be implemented to mitigate these risks. Future systems could embed proactive and explainable support mechanisms that help teachers verify, interpret, and improve AI-generated content as part of their natural workflow. For instance, rather than merely flagging the uncertainty of LLM-generated content, systems could use non-LLM-based verification pipelines to retrieve authoritative references or validated assessment banks to cross-check for correctness [49].

Another possibility is to make AI reasoning more transparent and inspectable. Instead of treating the model's internal process as

a black box, tools could display their reasoning process or confidence explanations that allow teachers to interrogate the rationale behind generated content [39, 44]. For instance, systems could show the inferred learning objective, reasoning steps, or aligned curriculum tags that led to its output to give teachers concrete points for critique or correction. Such features could turn validation from a cognitive burden into an interactive process of sense making. For teachers who coauthor assessments with colleagues, systems could also explore collaborative verification workflows. Rather than asking each teacher to independently judge AI outputs, they could allow peer co-review or lightweight consensus-building interfaces that pool teacher feedback on the generated assessments. Such distributed review processes could surface common errors and strengthen the collective reliability of AI-generated materials across teachers and schools. Designing for different risks means acknowledging that while AI can be useful for some teachers, it also carries uneven vulnerabilities. Building effective assessment authoring tools therefore requires equipping users—especially those under the most constraints—with support that match their needs.

8.3 Multilevel Reusable Interactions Demonstrate Promise but Require Deeper Evaluation

Evidence from codesign and the user study suggests that Ripplet's multilevel reusable interactions support the iterative dual process in our conceptual model (Section 4.1 and Section 5.4). Teachers used reusable commands to revise assessments, and several teachers (e.g., P8, U8) noted that this feature saved substantial time during assessment authoring (Section 7.5 and Section 6.2). However, we did not obtain objective measures of the effectiveness of multilevel reusable interactions (Section 7.3). Future work could include controlled A/B testing studies comparing authoring workflows with and without the command panel. Such experiments could quantify changes in authoring time, assessment quality, and teachers' experiences. Longitudinal studies could also potentially reveal whether reusable edits become more valuable as teachers accumulate a personalized library of commands over time. Finally, future work could investigate what types of edits multilevel reusable interactions are particularly well-suited for and where they fall short. Classifying the strengths, limitations, and failure modes of reusable edits could inform new forms of interaction designs that help teachers understand, trust, and more effectively reuse these commands. Such insights would also guide improvements to the underlying interaction paradigm, ensuring that multilevel reusable edits genuinely support the breadth of assessment authoring tasks teachers perform.

8.4 Effective Codesign Requires Aligning with Teachers' Schedule and Supporting their Growth

Engaging teachers in a seven-month codesign process across two semesters proved invaluable for developing Ripplet, and it also surfaced lessons about both the opportunities and challenges of sustained collaboration. Many teachers began with limited prior exposure to AI tools. Through the codesign process, they developed a better understanding of the probabilistic nature of these systems and learned strategies to frame prompts and manage expectations. In this way, codesign was not just about providing feedback on Ripplet, but also an opportunity for teachers themselves to build new knowledge and skills around working with AI.

At the same time, sustaining participation was demanding. Longitudinal codesign required repeated follow-ups and building trust and relationships. Teachers' workloads left little time to devote to research activities, and one partner remarked that they wished they had "*more time to play around with the tool outside* [*of the scheduled Zoom sessions*]." In addition, we timed Phase II to span a semester and Phase III to be when teachers are writing final exams so that teachers could meaningfully test Ripplet during periods when they were actively teaching and developing assessments (Figure 2). It is important to not only create intentional opportunities for open-ended exploration outside structured sessions, but also carefully align system design and evaluation periods with the rhythms of teachers' academic calendars.

We also observed differences in how teachers articulated design feedback. One partner with prior training in design thinking provided specific, actionable suggestions, while another partner defaulted to broader descriptors such as "user friendly" or "hard to use." Although all input was valuable, this variation suggests the opportunity for lightweight training to help codesign partners express more concrete, design-oriented feedback, without adding extra burden. Prior work similarly shows that even minimal, structured guidance can help people produce more specific and actionable design critiques [40]. Building reciprocal, sustainable, and effective codesign practices requires structuring collaborations that honor teachers' time and rhythms, while also creating space for their growth as designers and learners.

9 Conclusion

To support educational assessment authoring, we developed a conceptual model of teachers' workflows and a web-based system for authoring assessments through multilevel reusable interactions with LLMs. Over seven months, we codesigned Ripplet with 13 teachers to (1) develop a conceptual model of their assessment authoring practices and derive a set of design objectives; (2) build a prototype and refine the interactions with teachers to reach the final version of Ripplet, which supports generating and reusing questions from diverse inputs, adapting questions at multiple granularities, restructuring assessments, as well as tracking and reapplying teachers' edits; and (3) create assessments for their students and observe their evolved usage and behavior over time. We found that Ripplet enabled teachers to create formative assessments they would not have otherwise made, shifted their practices from generation to curation, and encouraged their reflection on assessment quality. In a user study with 15 additional teachers, we compared Ripplet with their current practice. Teachers felt the results were worth the effort more and the assessment quality improved (+1.93 and +1.11 on a 10-point scale, $p < 0.05$). Together, we demonstrate Ripplet as a valuable educational tool that improves both teacher experience and assessment quality through its multilevel reusable interaction paradigm.

Author Contributions

Yuan Cui conceptualized and led the project, conducted all co-design sessions, designed and implemented the system, analyzed the data, and drafted the manuscript. Annabel Marie Goldman contributed to system implementation, helped with user study, and drafted part of the manuscript. Jovy Zhou, Xiaolin Liu, Clarissa M. Shieh, Joshua Yao, and Mia Lillian Change contributed to system implementation. Matthew Kay conceptualized the project, supervised the work, and edited the manuscript. Fumeng Yang conceptualized and funded the project, supervised the work, contributed to system implementation, edited and drafted part of the manuscript.

Acknowledgments

This work would not have been possible without the generosity of our teacher partners in the codesign process and the user study. We are grateful for their time and invaluable feedback. We thank April Shi, Irena Liu, Laura Félix, Christopher Heo, Eric Lee, and Rachel Johnson for their early-stage contributions to building Ripplet. We extend our gratitude to Steven Moore, Mike Horn, Eleanor O'Rourke, and Duri Long for their feedback.

References

[1] 2020. Capturing Greater Context for Question Generation. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 05 (Apr 2020), 9065–9072. doi:10.1609/aaai.v34i05.6440

[2] Lenore Adie. 2013. The Development of Teacher Assessment Identity through Participation in Online Moderation. *Assessment in Education: Principles, Policy & Practice* 20, 1 (2013), 91–106. doi:10.1080/0969594X.2011.650150

[3] American Educational Research Association, American Psychological Association, and National Council on Measurement in Education. 2014. *Standards for Educational and Psychological Testing.* American Educational Research Association.

[4] Nicole Barnes, Helenrose Fives, and Charity M. Dacey. 2017. U.S. teachers' conceptions of the purposes of assessment. *Teaching and Teacher Education* 65 (2017), 107–116. doi:10.1016/j.tate.2017.02.017

[5] Paul Black and Dylan Wiliam. 1998. *Inside the black box: Raising standards through classroom assessment.* Granada Learning.

[6] Paul Black and Dylan Wiliam. 2009. Developing the theory of formative assessment. *Educational Assessment, Evaluation and Accountability (formerly: Journal of Personnel Evaluation in Education)* 21, 1 (2009), 5–31. 10.1007/s11092-008-9068-5.

[7] College Board. 2025. Advance Placement (AP) Program. https://ap.collegeboard.org/.

[8] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology* 3, 2 (2006), 77–101. doi:10.1191/1478088706qp063oa

[9] Gavin T. L. Brown. 2004. Teachers' conceptions of assessment: implications for policy and professional development. *Assessment in Education: Principles, Policy & Practice* 11, 3 (2004), 301–318.

[10] Gavin T. L. Brown. 2006. Teachers' Conceptions of Assessment: Validation of an Abridged Version. *Psychological Reports* 99, 1 (2006), 166–170. doi:10.2466/pr0.99.1.166-170

[11] Sally Brown. 2005. Assessment for learning. *Learning and teaching in higher education* 1 (2005), 81–89.

[12] Sahan Bulathwela, Hamze Muse, and Emine Yilmaz. 2023. Scalable Educational Question Generation with Pre-trained Language Models. In *Artificial Intelligence in Education*. 327–339.

[13] Yiming Cao, Zhen Li, Lizhen Cui, and Chunyan Miao. 2025. Adaptive Human-LLMs Interaction Collaboration: Reinforcement Learning driven Vision-Language Models for Medical Report Generation. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25)*. Association for Computing Machinery, New York, NY, USA, Article 62, 6 pages. doi:10.1145/3706599.3719852

[14] James Carifio and Rocco Perla. 2008. Resolving the 50-year debate around using and misusing Likert scales. *Med Educ* 42, 12 (Dec 2008), 1150–1152. doi:10.1111/j.1365-2923.2008.03172.x

[15] Guanliang Chen, Jie Yang, Claudia Hauff, and Geert-Jan Houben. 2018. LearningQ: A Large-Scale Dataset for Educational Question Generation. *Proceedings of the International AAAI Conference on Web and Social Media* 12, 1 (2018). doi:10.1609/icwsm.v12i1.14987

[16] Yi Cheng, Siyao Li, Bang Liu, Ruihui Zhao, Sujian Li, Chenghua Lin, and Yefeng Zheng. 2021. Guiding the Growth: Difficulty-Controllable Question Generation through Step-by-Step Rewriting. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*. 5968–5978. doi:10.18653/v1/2021.acl-long.465

[17] Zirui Cheng, Jingfei Xu, and Haojian Jin. 2024. TreeQuestion: Assessing Conceptual Learning Outcomes with LLM-Generated Multiple-Choice Questions. *Proceedings of the ACM on Human-Computer Interaction* 8, Article 431 (Nov. 2024), 29 pages. doi:10.1145/3686970

[18] Erin Cherry and Celine Latulipe. 2014. Quantifying the Creativity Support of Digital Tools through the Creativity Support Index. *ACM Trans. Comput.-Hum. Interact.* 21, 4, Article 21 (June 2014), 25 pages. doi:10.1145/2617588

[19] Ronald Jay Cohen, W. Joel Schneider, and Renée Tobin. 2022. *Psychological Testing and Assessment* (10th ed.). McGraw Hill LLC, New York, NY.

[20] Andrew Coombs, Christopher DeLuca, Danielle LaPointe-McEwan, and Agnieszka Chalas. 2018. Changing approaches to classroom assessment: An empirical study across teacher career stages. *Teaching and Teacher Education* 71 (2018), 134–144. doi:10.1016/j.tate.2017.12.010

[21] Yuan Cui, Lily W. Ge, Yiren Ding, Lane Harrison, Fumeng Yang, and Matthew Kay. 2025. Promises and Pitfalls: Using Large Language Models to Generate Visualization Items. *IEEE Transactions on Visualization and Computer Graphics* 31, 1 (2025), 1094–1104. doi:10.1109/TVCG.2024.3456309

[22] Christopher DeLuca and Don A. Klinger. 2010. Assessment literacy development: identifying gaps in teacher candidates' learning. *Assessment in Education: Principles, Policy & Practice* 17, 4 (2010), 419–438. doi:10.1080/0969594x.2010.516643

[23] Christopher DeLuca, Danielle LaPointe-McEwan, and Ulemu Luhanga. 2016. Teacher assessment literacy: A review of international standards and measures. *Educational Assessment, Evaluation and Accountability* 28, 3 (2016), 251–272. doi:10.1007/s11092-015-9233-6

[24] Steven M. Downing. 2005. The Effects of Violating Standard Item Writing Principles on Tests and Students: The Consequences of Using Flawed Test Items on Achievement Examinations in Medical Education. *Advances in Health Sciences Education* 10, 2 (2005), 133–143. doi:10.1007/s10459-004-4019-5

[25] Sabina Elkins, Ekaterina Kochmar, Iulian Serban, and Jackie C. K. Cheung. 2023. How Useful Are Educational Questions Generated by Large Language Models? *Communications in Computer and Information Science* (2023), 536–542. doi:10.1007/978-3-031-36336-8_83

[26] Haoxiang Fan, Guanzheng Chen, Xingbo Wang, and Zhenhui Peng. 2024. LessonPlanner: Assisting Novice Teachers to Prepare Pedagogy-Driven Lesson Plans with Large Language Models. Article 146, 20 pages. doi:10.1145/3654777.3676390

[27] K. J. Kevin Feng, Q. Vera Liao, Ziang Xiao, Jennifer Wortman Vaughan, Amy X. Zhang, and David W. McDonald. 2025. Canvil: Designerly Adaptation for LLM-Powered User Experiences. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 932, 22 pages. doi:10.1145/3706598.3713139

[28] Suzanne Fergus, Michelle Botha, and Mehrnoosh Ostovar. 2023. Evaluating Academic Answers Generated Using ChatGPT. *Journal of Chemical Education* 100, 4 (04 2023), 1672–1675.

[29] Sandra Ferketich. 1991. Focus on psychometrics. Aspects of item analysis. *Research in Nursing & Health* 14, 2 (1991), 165–168. doi:10.1002/nur.4770140211

[30] Yifan Gao, Lidong Bing, Wang Chen, Michael R Lyu, and Irwin King. 2018. Difficulty Controllable Generation of Reading Comprehension Questions. *Proceedings of the International Joint Conference on Artificial Intelligence* (2018). doi:10.48550/arxiv.1807.03586

[31] John Gardner. 2006. Assessment for learning: A compelling conceptualization. *Assessment and learning* (2006), 197–204.

[32] Abdallah Ghaicha. 2016. Theoretical Framework for Educational Assessment: A Synoptic Review. *Journal of Education and Practice* 7, 24 (2016), 212–231.

[33] Huanli Gong, Liangming Pan, and Hengchang Hu. 2022. KHANQ: A Dataset for Generating Deep Questions in Education. In *Proceedings of the 29th International Conference on Computational Linguistics*, Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na (Eds.). International Committee on Computational Linguistics, Gyeongju, Republic of Korea, 5925–5938. https://aclanthology.org/2022.coling-1.518/

[34] Thomas M Haladyna and Michael C Rodriguez. 2013. *Developing and Validating Test Items.* Routledge. doi:10.4324/9780203850381

[35] Ching Nam Hang, Chee Wei Tan, and Pei-Duo Yu. 2024. MCQGen: A Large Language Model-Driven MCQ Generator for Personalized Learning. *IEEE Access* 12 (2024), 102261–102273. doi:10.1109/ACCESS.2024.3420709

[36] Forrest Huang, Gang Li, Tao Li, and Yang Li. 2024. Automatic Macro Mining from Interaction Traces at Scale. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 1038, 16 pages. doi:10.1145/3613904.3642074

[37] Wenhui Kang, Lin Zhang, Xiaolan Peng, Hao Zhang, Anchi Li, Mengyao Wang, Jin Huang, Feng Tian, and Guozhong Dai. 2025. TutorCraftEase: Enhancing Pedagogical Question Creation with Large Language Models. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. Article 1076, 22 pages. doi:10.1145/3706598.3713731

[38] KaTeX Contributors. 2013. KaTeX. https://katex.org/.

[39] Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. 2020. Interpreting Interpretability: Understanding Data Scientists' Use of Interpretability Tools for Machine Learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–14. doi:10.1145/3313831.3376219

[40] Markus Krause, Tom Garncarz, JiaoJiao Song, Elizabeth M. Gerber, Brian P. Bailey, and Steven P. Dow. 2017. Critique Style Guide: Improving Crowdsourced Design Feedback with a Natural Language Model. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 4627–4639. doi:10.1145/3025453.3025883

[41] Stefan Küchemann, Steffen Steinert, Natalia Revenga, Matthias Schweinberger, Yavuz Dinc, Karina E. Avila, and Jochen Kuhn. 2023. Can ChatGPT support prospective teachers in physics task development? *Physical Review Physics Education Research* 19 (2023), 020128. Issue 2. doi:10.1103/PhysRevPhysEducRes.19.020128

[42] Ghader Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. 2020. A Systematic Review of Automatic Question Generation for Educational Purposes. *International Journal of Artificial Intelligence in Education* 30, 1 (2020), 121–204. doi:10.1007/s40593-019-00186-y

[43] Hui Min Lee, Peixin Hua, Rehab Alayoubi, and S. Shyam Sundar. 2025. Shorter and Simpler: Customizing Generative AI Responses Can Increase Satisfaction, Credibility, and Fact-Checking Intentions. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25)*. Association for Computing Machinery, New York, NY, USA, Article 479, 7 pages. doi:10.1145/3706599.3719749

[44] Brian Y. Lim, Anind K. Dey, and Daniel Avrahami. 2009. Why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 2119–2128. doi:10.1145/1518701.1519023

[45] Jintao Ling and Muhammad Afzaal. 2024. Automatic question-answer pairs generation using pre-trained large language models in higher education. *Computers and Education: Artificial Intelligence* 6 (2024), 100252. doi:10.1016/j.caeai.2024.100252

[46] Anne Looney, Joy Cumming, Fabienne van Der Kleij, and Karen Harris and. 2018. Reconceptualising the role of teachers as assessors: teacher assessment identity. *Assessment in Education: Principles, Policy & Practice* 25, 5 (2018), 442–467. doi:10.1080/0969594X.2016.1268090

[47] Xinyi Lu, Simin Fan, Jessica Houghton, Lu Wang, and Xu Wang. 2023. ReadingQuizMaker: A Human-NLP Collaborative System that Supports Instructors to Design High-Quality Reading Quiz Questions. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, Article 454, 18 pages. doi:10.1145/3544548.3580957

[48] Andrés Lucero. 2015. Using Affinity Diagrams to Evaluate Interactive Prototypes. In *Human-Computer Interaction – INTERACT 2015*, Julio Abascal, Simone Barbosa, Mirko Fetter, Tom Gross, Philippe Palanque, and Marco Winckler (Eds.). Springer International Publishing, Cham, 231–248.

[49] Harsh Maheshwari, Srikanth Tenneti, and Alwarappan Nakkiran. 2025. CiteFix: Enhancing RAG Accuracy Through Post-Processing Citation Correction. arXiv:2504.15629 [cs.IR] https://arxiv.org/abs/2504.15629

[50] Subhankar Maity, Aniket Deroy, and Sudeshna Sarkar. 2024. Exploring the capabilities of prompted large language models in educational and assessment applications. *arXiv preprint arXiv:2405.11579* (2024).

[51] Subhankar Maity, Aniket Deroy, and Sudeshna Sarkar. 2024. Harnessing the Power of Prompt-based Techniques for Generating School-Level Questions using Large Language Models. In *Proceedings of the Annual Meeting of the Forum for Information Retrieval Evaluation*. 30–39. doi:10.1145/3632754.3632755

[52] Subhankar Maity, Aniket Deroy, and Sudeshna Sarkar. 2024. How Effective is GPT-4 Turbo in Generating School-Level Questions from Textbooks Based on Bloom's Revised Taxonomy? *arXiv* (2024). doi:10.48550/arxiv.2406.15211

[53] Subhankar Maity, Aniket Deroy, and Sudeshna Sarkar. 2024. A Novel Multi-Stage Prompting Approach for Language Agnostic MCQ Generation Using GPT. In *Advances in Information Retrieval*. 268–277.

[54] Damien Masson, Sylvain Malacria, Géry Casiez, and Daniel Vogel. 2024. DirectGPT: A Direct Manipulation Interface to Interact with Large Language Models. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, Article 975, 16 pages. doi:10.1145/3613904.3642462

[55] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. 2019. Reliability and Inter-rater Reliability in Qualitative Research: Norms and Guidelines for CSCW and HCI Practice. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 72 (Nov. 2019), 23 pages. doi:10.1145/3359174

[56] James H McMillan. 1999. Establishing High Quality Classroom Assessments. (1999).

[57] James H. McMillan. 2003. Understanding and Improving Teachers' Classroom Assessment Decision Making: Implications for Theory and Practice. *Educational Measurement: Issues and Practice* 22, 4 (2003), 34–43. 10.1111/j.1745-3992.2003.tb00142.x.

[58] Craig A. Mertler. 2009. Teachers' assessment knowledge and their perceptions of the impact of classroom assessment professional development. *Improving Schools* 12, 2 (2009), 101–113.

[59] Steven Moore, Huy A. Nguyen, Norman Bier, Tanvi Domadia, and John Stamper. 2022. Assessing the Quality of Student-Generated Short Answer Questions Using GPT-3. In *Educating for a New Future: Making Sense of Technology-Enhanced Learning Adoption*. 243–257.

[60] Nikahat Mulla and Prachi Gharpure. 2023. Automatic question generation: a review of methodologies, datasets, evaluation metrics, and applications. *Progress in Artificial Intelligence* 12, 1 (2023), 1–32. doi:10.1007/s13748-023-00295-9

[61] Geoff Norman. 2010. Likert scales, levels of measurement and the "laws" of statistics. *Adv Health Sci Educ Theory Pract* 15, 5 (Dec 2010), 625–632. doi:10.1007/s10459-010-9222-y

[62] Liangming Pan, Yuxi Xie, Yansong Feng, Tat-Seng Chua, and Min-Yen Kan. 2020. Semantic Graphs for Generating Deep Questions. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. 1463–1475. doi:10.18653/v1/2020.acl-main.135

[63] Serafina Pastore and Heidi L. Andrade. 2019. Teacher assessment literacy: A three-dimensional model. *Teaching and Teacher Education* 84 (2019), 128–138.

[64] James W. Pellegrino, Naomi Chudowsky, and Robert Glaser (Eds.). 2001. *Knowing What Students Know: The Science and Design of Educational Assessment*. The National Academies Press, Washington, DC. doi:10.17226/10019

[65] PrismJS Contributors. 2012. PrismJS. https://prismjs.com/.

[66] Vatsal Raina and Mark Gales. 2022. Multiple-choice question generation: Towards an automated assessment framework. *arXiv preprint arXiv:2209.11830* (2022).

[67] React Syntax Highlighter Contributors. 2016. React Syntax Highlighter. https://github.com/react-syntax-highlighter/react-syntax-highlighter.

[68] rehype Contributors. 2017. rehype-katex. https://www.npmjs.com/package/rehype-katex.

[69] remarkjs Contributors. 2016. remark-math. https://github.com/remarkjs/remark-math.

[70] Ana Remesal. 2011. Primary and secondary teachers' conceptions of assessment: A qualitative study. *Teaching and Teacher Education* 27, 2 (2011), 472–482. 10.1016/j.tate.2010.09.017.

[71] Mohi Reza, Ioannis Anastasopoulos, Shreya Bhandari, and Zachary A. Pardos. 2025. PromptHive: Bringing Subject Matter Experts Back to the Forefront with Collaborative Prompt Engineering for Educational Content Creation. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 148, 22 pages. doi:10.1145/3706598.3714051

[72] Mohi Reza, Nathan M Laundry, Ilya Musabirov, Peter Dushniku, Zhi Yuan "Michael" Yu, Kashish Mittal, Tovi Grossman, Michael Liut, Anastasia Kuzminykh, and Joseph Jay Williams. 2024. ABScribe: Rapid Exploration & Organization of Multiple Writing Variations in Human-AI Co-Writing Tasks using Large Language Models. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 1042, 18 pages. doi:10.1145/3613904.3641899

[73] Bahare Riahi and Veronica Cateté. 2025. *Comparative Analysis of STEM and Non-STEM Teachers' Needs for Integrating AI into Educational Environments*. Springer Nature Switzerland, 125–140. doi:10.1007/978-3-031-93567-1_9

[74] Tim Steuer, Leonard Bongard, Jan Uhlig, and Gianluca Zimmer. 2021. On the Linguistic and Pedagogical Quality of Automatic Question Generation via Neural Machine Translation. In *Proceedings of the European Conference on Technology Enhanced Learning*. 289–294. doi:10.1007/978-3-030-86436-1_22

[75] Richard J Stiggins. 1988. Revitalizing classroom assessment: The highest instructional priority. *The Phi Delta Kappan* 69, 5 (1988), 363–368. https://www.jstor.org/stable/20403636.

[76] Richard J. Stiggins. 2001. The Unfulfilled Promise of Classroom Assessment. *Educational Measurement: Issues and Practice* 20, 3 (2001), 5–15.

[77] Richard J. Stiggins, Nancy F. Conklin, and Nancy J. Bridgeford. 1986. Classroom assessment: A key to effective education. *Educational Measurement: Issues and Practice* 5, 2 (1986), 5–17. 10.1111/j.1745-3992.1986.tb00473.x.

[78] John M. Stonehouse and Guy J. Forrester. 1998. Robustness of the t and U tests under combined assumption violations. *Journal of Applied Statistics* 25, 1 (1998), 63–74. arXiv:https://doi.org/10.1080/02664769823304 doi:10.1080/02664769823304

[79] Sue Swaffield. 2008. *Unlocking assessment: Understanding for reflection and application*. Routledge.

[80] Zachari Swiecki, Hassan Khosravi, Guanliang Chen, Roberto Martinez-Maldonado, Jason M. Lodge, Sandra Milligan, Neil Selwyn, and Dragan Gašević.

2022. Assessment in the age of artificial intelligence. *Computers and Education: Artificial Intelligence* 3 (2022), 100075. doi:10.1016/j.caeai.2022.100075

[81] Maddalena Taras. 2009. Summative assessment: The missing link for formative assessment. *Journal of Further and Higher Education* 33, 1 (2009), 57–69. 10.1080/03098770802638671.

[82] Marie Tarrant, Aimee Knierim, Sasha K. Hayes, and James Ware. 2006. The frequency of item writing flaws in multiple-choice questions used in high stakes nursing assessments. *Nurse Education Today* 26, 8 (2006), 662–671. doi:10.1016/j.nedt.2006.07.006

[83] Robin D Tierney. 2012. Fairness in classroom assessment. *SAGE handbook of research on classroom assessment* (2012), 125–145.

[84] Luu Anh Tuan, Darsh Shah, and Regina Barzilay. 2020. Capturing Greater Context for Question Generation. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 05 (2020), 9065–9072. doi:10.1609/aaai.v34i05.6440

[85] Masaki Uto, Yuto Tomikawa, and Ayaka Suzuki. 2023. Difficulty-Controllable Neural Question Generation for Reading Comprehension using Item Response Theory. In *Proceedings of the Workshop on Innovative Use of NLP for Building Educational Applications*. 119–129. doi:10.18653/v1/2023.bea-1.10

[86] Veronica Villarroel, Susan Bloxham, Daniela Bruna, Carola Bruna, and Constanza Herrera-Seda. 2018. Authentic Assessment: Creating A Blueprint for Course Design. *Assessment & Evaluation in Higher Education* 43, 5 (2018), 840–854. doi:10.1080/02602938.2017.1412396

[87] Jianhui Wang, Yangfan He, Yan Zhong, Xinyuan Song, Jiayi Su, Yuheng Feng, Ruoyu Wang, Hongyang He, Wenyu Zhu, Xinhang Yuan, Miao Zhang, Keqin Li, Jiaqi Chen, Tianyu Shi, and Xueqian Wang. 2025. Twin Co-Adaptive Dialogue for Progressive Image Generation. In *Proceedings of the 33rd ACM International Conference on Multimedia* (Dublin, Ireland) *(MM '25)*. Association for Computing Machinery, New York, NY, USA, 3645–3653. doi:10.1145/3746027.3755141

[88] Xu Wang, Simin Fan, Jessica Houghton, and Lu Wang. 2022. Towards Process-Oriented, Modular, and Versatile Question Generation that Meets Educational Needs. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 291–302. 10.18653/v1/2022.naacl-main.22.

[89] Zichao Wang, Andrew S. Lan, Weili Nie, Andrew E. Waters, Phillip J. Grimaldi, and Richard G. Baraniuk. 2018. QG-net: a data-driven question generation model for educational content. In *Proceedings of the Annual ACM Conference on Learning at Scale*. Article 7, 10 pages. doi:10.1145/3231644.3231654

[90] WestEd. 2025. The Next Generation Science Standards. https://www.nextgenscience.org/.

[91] Nico Willert and Jonathan Thiemann. 2024. Template-Based Generator for Single-Choice Questions. *Technology, Knowledge and Learning* 29, 1 (2024), 355–370. doi:10.1007/s10758-023-09659-5

[92] Chris Woolston. 2015. Psychology journal bans P values. *Nature* 519, 7541 (2015), 9–9. doi:10.1038/519009f

[93] Yuxi Xie, Liangming Pan, Dongzhe Wang, Min-Yen Kan, and Yansong Feng. 2020. Exploring Question-Specific Rewards for Generating Deep Questions. In *Proceedings of the International Conference on Computational Linguistics*. 2534–2546. doi:10.18653/v1/2020.coling-main.228

[94] Yueting Xu and Gavin T.L. Brown. 2016. Teacher assessment literacy in practice: A reconceptualization. *Teaching and Teacher Education* 58 (2016), 149–162. doi:10.1016/j.tate.2016.05.010

[95] Ryan Yen, Jiawen Stefanie Zhu, Sangho Suh, Haijun Xia, and Jian Zhao. 2024. CoLadder: Manipulating Code Generation via Multi-Level Blocks. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 11, 20 pages. doi:10.1145/3654777.3676357

[96] Yaxuan Yin, Shamya Karumbaiah, and Shona Acquaye. 2025. Responsible AI in Education: Understanding Teachers' Priorities and Contextual Challenges. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency (FAccT '25)*. Association for Computing Machinery, New York, NY, USA, 2705–2727. doi:10.1145/3715275.3732176

[97] Guanhua Zhang, Mohamed Adel Naguib Ahmed, Zhiming Hu, and Andreas Bulling. 2025. SummAct: Uncovering User Intentions Through Interactive Behaviour Summarisation. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 265, 17 pages. doi:10.1145/3706598.3713190