Generative AI Tutor: Supporting Students' Critical Reflection in Assignment Revision

Laura Sinikallio[1,*], Lili Aunimo[2] and Tomi Männistö[1]

[1]University of Helsinki, Helsinki, Finland
[2]Haaga-Helia University of Applied Sciences, Helsinki, Finland

Abstract

Generative AI has surged into education. At the same time, a heated debate on the risks and benefits of GenAI is ongoing; many fear that thinking is being outsourced. We investigated ways to harness GenAI for (a) teaching research methods and research planning at the master's degree level, and (b) supporting students' reflective thinking and argumentation skills. We created a two-phase course assignment framework consisting of an initial student-written submission, followed by a revision round guided by a custom-prompted virtual tutor based on retrieval-augmented generation and GPT. We emphasized guidance over direct answers and pushed for sound argumentation. The students found working with the virtual tutor mostly helpful and useful, but lamented the tutor's wordiness and occasional shaky understanding of context. We noted that the modifications made to the assignments after interacting with the virtual tutor had a positive effect on the grading and improved the quality of the final submissions. In several cases, the argumentation behind the claims was improved after interacting with the virtual tutor. Education is approaching a global turning point. We urgently need educational systems that keep students thinking and reflecting. In this paper, we propose an approach for implementing GenAI in education in a role that supports reflection rather than eliminating the need for it. We showcase potential uses and considerations of GenAI that may benefit educators.

Keywords

Generative AI, virtual tutor, critical reflection, student perception, prompt engineering, human-AI dialogue

1. Introduction

Generative AI (GenAI) tools have recently surged into education, brought in by both teachers *and* students [1]. To realize the full potential of these tools or to mitigate their negative effects, it is essential to understand their impact on pedagogy. Education evolves as new technologies become available, but their mere adoption does not ensure favorable outcomes [2]. The impact of GenAI, or large language model-based (LLM) tools, is still largely unknown [3]. Several studies show that students use GenAI widely in any case, see, e.g., [4, 5]. It is important to provide students with pedagogically sound ways to use them.

The capabilities of GenAI tools in education have attracted significant academic interest for a couple of years now. At least in the context of computer science education, most studies have initially emphasized tool capabilities and performance benchmarks, with less focus on how students actually use and interact with them [1]. Lately, however, student interaction with GenAI has seen more research published (e.g. [3, 6, 7, 8]). These studies often focus on the teaching of introductory programming and novices, often freshmen. Other CS topics and student levels have received much less attention.

At the other end of the spectrum are master's students who are deepening their understanding of not only the topics of their field but also their academic skills: research, and scientific and professional communication. All of these require the skills of deliberate, reflective, and reasoned action, important to any professional in any field. A central tool behind these skills is argumentation, the process (and skill) of reasoning systematically [9]. We find that there is a growing need for educational systems that

NWISEd 2026: Workshop on Co-Creating New Ways of Information Systems Education, July 1–3, 2026, Swansea, United Kingdom
*Corresponding author.
laura.sinikallio@helsinki.fi (L. Sinikallio); lili.aunimo@haaga-helia.fi (L. Aunimo); tomi.mannisto@helsinki.fi (T. Männistö)
0000-0001-7398-6585 (L. Sinikallio); 0000-0002-3336-0948 (L. Aunimo); 0000-0001-7470-5183 (T. Männistö)
© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-11-19

CEUR-WS.org/Vol-4333/paper6.pdf

support students in practicing these skills; systems that make students think. Lately, another related question has emerged: How to implement GenAI in a role that supports this educational goal rather than removing the need for it?

We present in this research a proposal for implementing GenAI in education in a role that supports the practice of argumentation and reflection. We support our approach with an field study in the context of master's level software engineering research methodology education. We designed a coursework in which students iteratively work on their assignments (mock-up research plans), first on their own, then with feedback and guidance from a custom-prompted GenAI tutor. Previous research shows that academic achievement is often closely linked to students' interactions with the teacher and the feedback they receive [10]. Thus, we hypothesize that interacting with a virtual tutor and receiving feedback from it might at least partly provide the same effect.

We aimed to create a setup that offered students ample practice in academic writing, which is arguably often in a very small role in CS education compared to many other academic disciplines. Secondly, both research and software engineering (as well as most CS tracks) are iterative processes; projects are worked on and reworked multiple times. Iterating is a valuable approach in itself and becomes easier with practice. Our framework leads students to return to their work and think it through again, deepening their understanding of the topic. The availability of a virtual tutor can make revisiting one's work more motivating; instead of just staring at their own work, the students have impartial, on-demand feedback available when they need it.

As research terminology can occasionally be confusing, and papers used as examples often assume some level of background, students in the research methodology course had previously reported some difficulties understanding all the concepts. Some students might feel more comfortable receiving feedback and asking questions from a virtual entity rather than from human tutors [11]. GenAI tools may support students in reaching learning objectives even when they do not really know what to ask; the tools may provide accessible explanations and examples even when posed vague queries [6].

Our research questions are as follows:

RQ1. What are students' and instructors' perceptions of the feasibility and usefulness of employing GenAI for support in learning research planning?

RQ2. To what extent does a customized GenAI tutor improve the quality of students' research plans?

RQ3. What kinds of human-AI dialogue strategies do students use while aiming to improve their research plans?

RQ4. Do some human-AI dialogue strategies correlate with assignment success?

With RQ1, we wanted to survey the general usability of this setting from both the students' and the instructors' side. The research question was informed by the Technology Acceptance Model (TAM), which states that perceived usefulness and ease of use determine system usage [12]. We also examined how the students' assignments improved after working with the virtual tutor (RQ2). We looked at received grades but especially at the quality of argumentation, the key meta skill behind well-conducted research, as we find that a sound argumentation of a topic portrays understanding of the topic itself. We also took a deeper look at the student - virtual tutor interaction (RQ3) and whether it correlated with course grades (RQ4).

2. Related work

In computer science, much of the emerging research on student-(Gen)AI interaction focuses on programming tasks. For example, both [6] and [7] conducted studies on how CS1 students leverage LLMs and interact with them for programming tasks. Both studies noted the common tendency of passing the productive struggle and problem ownership to LLMs. In [6], the researchers recognized two main student approaches to problem solving: iterative (active refining of prompts to achieve correct answers) and linear (using the full problem description as a prompt to find answers directly without personal

effort). The latter strategy was used by many participants and could be seen as an indicator of a tendency to seek quick fixes and a potential negative impact on students' learning. This aligns with the notions of [7], who noted that students often treated the LLM as an authority in a pseudo-apprenticeship, but without ever progressing to later stages of apprenticeship, such as scaffolded practice or articulation. Rather, they repeatedly observed behaviors such as reading without refining, bypassing productive struggle, and ceding problem ownership. Similar distributions have been noted by [13], who analyzed students' behavior when interacting with LLMs during code refactoring tasks. The two most common behavior types they found were delegating the entire refactoring task to the LLM, and only using the LLM to ask questions about their refactored code. They observed a relation between the latter type and solutions of good quality. In our research, we wish to mitigate the observed negative effects by limiting the availability of ready solutions by setting guardrails for the GenAI.

In contexts outside programming education, a recent study [14] investigated the use of RAG-based (retrieval-augmented generation) custom-prompted LLM chats as teaching assistants for medical students. The study showed that students found chatting with the virtual teaching assistants especially useful during the pre-exam period, when they needed reliable assistance quickly. An LLM-based virtual tutor has also been found beneficial in facilitating learning of argumentation skills. In an experiment by [15], students were assigned to one of two versions of an intelligent writing assistant: one based on a fine-tuned BERT classification model and the other on a custom-prompted GPT-3.5 API. Both intelligent writing assistants focused on providing feedback on the argumentative structure of the student's texts. The initial results showed that the GPT-based virtual tutor may be more effective in facilitating student learning. Computer-supported argumentation learning is a longstanding field [16], but the recent appearance and wide adoption of GenAI tools, such as ChatGPT, open new interesting opportunities that need to be studied further. In this paper, we describe one such investigation in the context of software engineering research education.

3. Methods

In this research, we wanted to see how iteratively working on research plans with a GenAI virtual tutor could help students improve their written research plans and thinking.

3.1. CurreChat – an educational GenAI tool

We used CurreChat, a GenAI-based tool developed at the University of Helsinki[1]. CurreChat has been designed specifically for teaching needs and is freely available to the staff of the University of Helsinki; however, token limits apply. CurreChat is built on OpenAI LLMs and, at the time of writing, offers four different models. All research described in this paper was conducted using the GPT-5 model, the latest model available at the time. CurreChat complies with the EU General Data Protection Regulation (GDPR). Conversations are private and are not stored in the system. Users can choose to save their conversations by sending the chat logs to themselves via email. The materials entered into the service are not used to train language models.

CurreChat is used via a web-page chat environment (though a Moodle embed became available later). The CurreChat UI is directly comparable to any other modern (GenAI) chat UI. University staff can use CurreChat freely in a neutral, non-course-context or in their course-specific instances. Staff can create either personal or course-context prompts and manage the course instance settings (prompt visibility, RAG rules, etc.) widely. For students, CurreChat availability depends on the courses they are enrolled in. If the teacher has enabled CurreChat for a course, students of that course can use the service for the duration of the course.

[1]https://github.com/UniversityOfHelsinkiCS/gptwrapper

3.2. Study setup

This study was conducted during a software engineering research methodology course. The course is optional and offered by the Computer Science Master's programme. The course covers several qualitative research methods and concepts that are common to SE research. It also aims to support students in conducting their master's thesis, so participants are often planning to start their own research or have already done so. As the new in-house AI tool, CurreChat, became available, a new set of assignments and a virtual tutoring setup were designed for this course to offer students new resources and test the viability of virtual tutoring. While the assignments in the course were heavily reinvented, we took care to ensure that students in this instance would not be offered (negatively) inferior chances or teaching compared to their cohorts in previous instances. We saw the changes in assignments as pedagogically sensible, and the addition of an AI tutor did not replace any pre-existing human tutoring but instead complemented previously available resources.

All course participants were informed of this research at the beginning of the course, both verbally and in writing. Before being able to submit any assignments, the students were required to read and answer (y/n) to a research permit form. This form stated the methods and purpose of the research, as well as information on data collection and handling. It was emphasized that participation was entirely voluntary and consent could be withdrawn at any time without any consequences. All course resources were available to all students regardless of their consent.

3.3. Refactoring assignment

During the course, the students practiced creating software engineering research questions and plans by writing short mock-up research plans according to the "research method of the week". These plans were to be written in the form of a *Methods* section of a thesis or an academic paper. The course assignments also included interactive discussion boards, a couple of other small text-based assignments, and a final extended study plan on a method and topic of their choosing. This research focuses on the mock-up research plans.

The weekly research plan assignments were done in two phases, as shown in Figure 1. The students would start Phase 1 by watching a pre-recorded lecture on the research method in focus – Systematic Literature Review, Action Research, or Design Science Research[2] – and read assigned papers. Then they would start their written assignment. The students were given a research scenario with some high-level research goals, and their task was to write a short (1-2 pages) research plan accordingly, following the research method of the week. These assignments were turned in before the in-person class discussion.

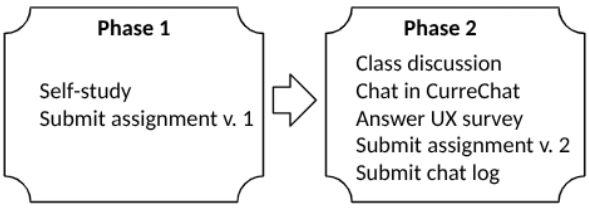


Figure 1: The progression of the weekly research plan assignments

In the classroom session, which marked the start of Phase 2 of the assignment, the students would discuss the reading materials in groups, answer questions, and add comments to a Miro[3] whiteboard prepared by the teacher. During the session, the teacher commented on the students' input in Miro, and at the end, a teacher-led discussion covered topics that arose from the board or students' questions. After the session, the assignment-specific CurreChat virtual tutor became available to students, with carefully designed prompts that also used the RAG system. The RAG system was given assignment-specific

[2]The course also covered Case Study, but due to technical difficulties, that week's data was left out of this research.
[3]https://miro.com/index/

inputs, so the source materials varied accordingly. The students could then ask the virtual tutor for feedback on the first version of their research plan and to chat with it. The students were free to revise their plans as much or as little as they wished. After chatting in CurreChat, the students were required to save their chat logs, complete a short survey about their user experiences, and summarize the changes they had made. After this, they were able to turn in the revised final version, along with their chat logs, and thus complete the week's assignments.

3.4. Virtual tutor prompts

To support students in their revision work, we created virtual tutors and prompts for each research plan assignment. These custom-prompted GenAI tutors were informed of the course and assignment context and told to focus on constructive, easy-to-understand feedback without providing ready answers. GenAI systems have been shown to encourage undesired levels of answer seeking behavior, but studying is not only about finding the solution but also about achieving learning objectives [6]. In our context, we wished to encourage the practice of reflecting on and justifying one's choices rather than just listing research action points.

Our main **design objectives** for the virtual tutors were as follows.

1. **Guidance-oriented interaction**. The tutor should not provide complete solutions (e.g., RQs), but scaffold the student by offering guiding questions, strategic hints and partial steps.
2. **Context-sensitivity**. The tutor should be aware of the assignments and course context, taking into consideration the scope and required elements.
3. **Accessibility for novices**. The tutor should avoid academic jargon where possible and provide clear comments and instructions that are comprehensible to a master's level student conducting their first proper research.
4. **Conciseness**. The tutor should be concise in its feedback and suggestions to make them easier to process and the chatting experience smoother.

All prompts for each tutor (assignment) followed the same structure and essentially the same contents. Each prompt was split into ten cross-referencing sections: *Role, Principles, Context, Guardrails, Capabilities, Flow Logic, Format Guide, Hidden Reasoning Rubric, Fail-Safe,* and *Self-Check.* Each section varied from one sentence to page length. The prompts were informed by structural examples from prompt generators and guides, but the contents were handwritten. Some assignment and research method details varied across prompts. After the first assignment cycle, notable modifications to answer-length limits were added. Limits had already existed on the answer length and structure of the tutor's generated answers, but it still proved to be cumbersomely verbose and inconsistent on that front.

The virtual tutors were explicitly prompted to pay attention to the argumentation structure in students' assignments and to guide students in improving their plans. Thus, the prompt included the following instructions: *Consider especially how well the study design has been explained and choices argued for* and *If any claims lack clear premises, point them out.* Additionally, we prompted the tutors to emphasize the Socratic method of guiding through questioning rather than providing direct hints (they easily got *too* helpful and mitigated the need for reflection).

3.5. Data collection

We collected various kinds of data. First, the first and final submitted versions of *the students' assignments* and *the grades* they received. Both versions were graded separately, with the first submission earning more points, to ensure proper effort before seeking help from the virtual tutor. Second, *the chat logs* from CurreChat. Lastly, the students also completed a *short survey* about their user experience working with CurreChat, providing quantitative data on student perceptions. The survey contained three sections: 1) seven Likert-scale statements described in the Results section, 2) a (required) text field for briefly explaining what had changed between the initial and final versions of the research plan, 3) an optional text field for sharing their thoughts about the virtual tutor. To encourage students to voice

their opinions, they were told that thoughtful analyses and comments would be "favorably noted" in their final grade, and that critical comments would not be penalized. The course was completed by 14 students, predominantly men, of whom 12 participated in this research. From the second week (action research), we miss data from one student.

4. Results

Next, we go over the results. The results are structured according to our research questions.

4.1. RQ1: Student and teacher perceptions

We wanted to know *what are students' and instructors' perceptions of the feasibility, and usefulness of using GenAI for support in learning research planning?* Each week, we asked the students for their opinions on the feedback they received from the CurreChat virtual tutor (understandability, usefulness, relevance, learning gain) and on their user experience (trust, ease of use, and cumbersomeness). The students were offered seven statements they answered on a five-point scale (1: strongly disagree, 3: I don't know, 5: strongly agree). The full statements and a summary of the survey results are shown in Table 1. Higher values indicate positive sentiment except for the last statement, where sentiment direction is inverted. Average values for the whole course are shown under *Total averages*, encompassing three weeks, or topics, and thus three sets of inputs from each student[4]. Under *Topic averages* are the averages for each week (topic) of the course (systematic literature survey, action research, design science research). The section *Sentiment across all topics* shows the absolute number of different answer sentiments. The sentiment inversion for S7 has been accounted for.

Overall, the students rated the CurreChat virtual tutors quite positively, with even the most negative perceptions averaging as high as 3.1 (S5, *trusting the given feedback*). For this statement, a lower value can also be considered a somewhat positive result; GenAI's tendency to hallucinate information is well known, so we appreciate the students' reservations. The second most negative average was for S7 (2.7, *CurreChat was more cumbersome to use than other similar tools*). This could, in some way, be affected by the guardrails we had set; the tutors would not give outright rewording or examples that could be copied and pasted as is. The scores stayed quite stable across the weeks, apart from the notable upward jump of 0.8 points between T1 and T2 for S1 (*easy to understand*). As stated earlier, we reinforced the length limitations on the prompt after the first topic and suspect it explains much of the positive change; some students specifically mentioned a positive change in their T2 survey.

In each survey, students also had the option to share their user experiences in their own words. Some students consistently raised concerns or dislikes about using GenAI. They questioned the environmental and ethical impacts and expressed concern that human feedback and thinking were being ousted. Some of these students reported feeling detached from this step of the assignment or actively disliking it. Most of the students reported positive or ambivalent experiences. Many stated that they had received good suggestions or useful feedback from the tutor that helped them see where their plans could improve. At the same time, there were many notes on the tutor producing too much text, expecting things out of scope or using too much "academese". Overall, the comments were still more positive than negative.

From the instructors' point of view, we found that the CurreChat GenAI could provide high-quality feedback out of the box or with near zero-shot prompting. Writing the tutor prompts very quickly became more about limiting the AI rather than adding things for it to do. The GenAI model was very eager to help, i.e., to give ready answers and examples, and was very verbose, producing walls of text full of academic jargon.

In general, we found the return on investment for the time spent prompting to be highly U-shaped. Initial prompts, for example, about not producing complete rewritten research questions, led the AI to avoid using even the students' own initial RQs, even though it was prompted to repeat them alongside feedback. Rather, it would produce sentences like: *[Imagine your RQ here].* Additionally, the AI often

[4]Bar one student from topic 2.

Table 1
Student perceptions across topics and statements (5-point Likert scale, 3: I don't know). Higher values indicate more positive evaluations (except for S7, where sentiment direction is inverted.)

Statements	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	*n*
				Total averages				
	3.7	3.8	3.8	3.5	3.1	4.2	2.7	*35*
				Topic averages				
T1 (SLR)	3.3	3.7	3.8	3.8	2.9	4.2	2.8	*12*
T2 (AR)	4.1	3.9	3.8	3.5	3.2	4.3	2.8	*11*
T3 (DSR)	3.9	3.8	3.7	3.3	3.2	4.2	2.5	*12*
				Sentiment counts across all topics				
Negative (< 3)	3	2	1	5	9	1	9	*35*
Positive (> 3)	32	23	23	20	13	28	13	*35*
I don't know (3)	9	10	11	10	13	6	13	*35*

Statements

S1: The feedback given / questions asked by the CurreChat Virtual Assistant (VA) were easy to understand

S2: The feedback given / questions asked by the VA were (to my understanding) relevant and meaningful

S3: The feedback and comments given and the questions asked by the VA were useful to my learning

S4: I feel that I learnt more with CurreChat than without

S5: I can trust the feedback given by CurreChat

S6: CurreChat was easy to use

S7: CurreChat was more cumbersome to use than other GenAI tools that I use

failed to stick to length limitations, nor was it consistent in its answers (grading, style, length, etc.). It might follow the feedback structure rules the first time the student asks for feedback, and then produce a lengthy freestyle feedback the next time, or it might first produce the required structure and then add additional sections after it. The prompting process focused on balancing harnessing GenAI without losing too much of its adaptive capacity, which was the central reason for adopting GenAI for this teaching scenario in the first place. Prompting the tutor to adopt the Socratic method (guiding by asking questions) was very helpful in this effort. While the AI might sometimes produce questions that were quite hint-rich in themselves (e.g., *Could you consider stating x or y in the context of u/w/z?*), they still required the student to piece the sentences together themselves.

4.2. RQ2: Text and argumentation improvement

We studied *to what extent a customized GenAI tutor improves the quality of students' research plans.* We paid special attention to the quality of argumentation in the assignments that had been considerably improved in the second phase. By greatest improvement, we refer to the assignments with the largest increase in the percentage of points gained between the phases (versions). Table 2 shows the percentage of points for each informant for the phases 1 and 2 and for all three tasks: systematic literature review (SLR), action research (AR) and design science research (DSR). The maximum number of points for the first phase was 6, and for the second phase, 4. The rationale behind having a higher maximum for phase 1 was to encourage the students to put more effort into the initial phase, where they worked without the help of the virtual tutor. The second phase was considered as an opportunity to improve the research plan, and points were given both on the quality of the final version and on the soundness of the changes made.

We observe from Table 2 that in each task there are students whose work improves considerably. Informants 6, 4, and 1 had big improvements in points in tasks SLR, AR and DSR, respectively. However, some students who already had good points in the first phase did not either improve the most notable shortcomings of their assignment meaningfully, or their changes introduced new issues, such as

Table 2
SLR, AR, and DSR assignment grades. Each grade is presented as the percentage of points gained out of the maximum. The assignments with the greatest rise in gained percentage between phases are marked in bold.

Informant ID	Percentage of points					
	SLR		AR		DSR	
	Phase 1	Phase 2	Phase 1	Phase 2	Phase 1	Phase 2
1	67	50	67	75	**50**	**100**
2	83	75	83	75	83	100
3	100	100	100	100	100	100
4	83	75	**67**	**100**	100	100
5	100	100	83	100	67	100
6	**50**	**100**	100	100	100	100
7	67	50	100	100	67	75
8	83	50	0	0	83	50
9	100	75	83	75	83	50
10	50	75	67	50	33	0
11	100	100	100	100	100	100
12	100	100	100	100	83	100
Average	81.92	79.17	79.17	81.25	79.08	81.25

contradictions in the text. This is the case for informants 9 (SLR), 2 (AR) and 8 (DSR).

We were especially interested in studying the changes in the amount and type of argumentation between the initial and revised research plans. We analyzed the argumentative structures in the two versions for the informants whose points had risen most between the two phases. The research plans of one informant per topic (SLR, AR and DSR) were analyzed: informants 6, 4 and 1, respectively. We classified the argumentative structures into two types: 1) *claim and premise* and 2) *claim, counter-argument, and rebuttal*. This division into two types is in line with the relation-extraction task that is common in the argument mining literature, where arguments are first extracted and segmented into claims and premises. Subsequently, the relation prediction task identifies the relation between the claim and the premise as support or attack [17]. Table 3 shows examples taken from the student assignments illustrating each type. The table also shows the number of argumentation types added in the second phase. We observe that, among the three informants, the total number of additions of argumentative structures was 14, and ten out of them had the simple structure *claim and premise*.

Table 3
One example of each of the two types of argumentation structures found in the assignments. The last column shows the total number of additions for each type in the three cases analysed.

Type	Example	# Additions
Claim Premise	It was selected because the tool needs to be fast enough so that the development process isn't slowed down.	10
Claim Counter-argument Rebuttal	Other quality-based filtering was not done, which can be considered a limitation, but it should not diminish the validity of the results as the purpose of this study is to understand the capabilities of LLMs instead of studying the usefulness of those capabilities.	4

4.3. RQ3: Student-GenAI dialogue strategies

We also took a look into *what kinds of human-AI dialogue strategies do students use while aiming to improve their research plans?* The discussions the students had with the virtual tutors varied greatly: Of the 35 separate chats, the shortest was just one student turn (i.e., the student just submitted their research plan for feedback), and the longest was 20 student turns. The shorter discussions were much more prevalent, 66% (n = 23) of all discussions were less than five student turns in length. Average lengths varied weekly/per topic: the average lengths for T1, T2, and T3 were 4.4, 6.4, and 3.3, respectively.

The average for T2 is elevated by two exceptionally long discussions, but overall, the values for T3 are notably lower than in previous weeks. There could be many causes for this: Perhaps the students became more weary toward the end of the course period (T3 was covered in the last week of teaching) and did not have the motivation to dig deeper into the feedback, or maybe their research planning skills or familiarity with the generated feedback had improved so that they required less feedback dialogue.

We identified three interaction strategies in the student chat logs: *passive checker*, *outsourcer*, and *engaged converser*. These strategies were not immutable; 33% of the students employed two different strategies across the weeks. The first strategy group, the passive checkers, did not engage the tutor in discussion. They mostly–or even only–submitted the entire research plan or a revised section with a minimal preface, such as *"Some changes"*, without asking any specific questions or commenting on the feedback. Their "discussion" lengths were typically only a few turns in length. Half of the students used this strategy either for the whole course or part of it.

Secondly, we recognized a few cases in which a student tried to outsource the plan refinement directly. They would ask the tutor to provide them with ready examples for a section. When the tutor would not fully comply[5], these students adopted another strategy. We saw a shift towards both the passive checkers and the engaged conversers. The outsourcing strategy was used by only 17% of the students.

In the third group–engaged conversers–the students engaged with the virtual tutorial by asking follow-up questions, requesting clarifications or suggestions for next actions. They would also argue with the tutor, question its feedback, and admonish it for obvious fallacies. These discussions had instances of "true" back-and-forth dialogue. We saw different types of engaged dialogue: some students were more confrontational, some provided more justifications for their comments, and some were "exploring", trying to understand the tutor's grading system, for example. Most students (67%) used the conversational strategy at least once during the course. Interestingly, we noted that some engaged conversers drifted towards passive checking towards the end of the course, but none of the initially passive checkers shifted into engaged conversers during the course.

4.4. RQ4: Correlation of grades and interaction strategies

When looking at the different interaction strategies the students employed, we wanted to see if there were any visible connections to grading (*Do some human-AI dialogue strategies correlate to assignment success?*). We noted that all of the top performers on the course had used the engaged converser strategy when working with the virtual tutors. On the other hand, the outsourcers and some passive checkers formed the lowest performing group. Students who had been passive checkers had both high and low grades. Assignments were graded before the analysis of AI-interaction, so blind to these categories. While the outsourcing and engaged conversation approaches could tentatively be proposed to predict lower and higher grades, respectively, similar correlations could not be observed for passive checkers. For passive checkers, the students' initial skills were likely the determining factor in their grades. When comparing the grades between the first and final versions of the students' research plans, the outsourcers and some of the passive checkers improved the least.

5. Discussion

In this paper, we propose an approach for implementing GenAI in education in a role that supports students' critical reflection. We share the results from our first use case of this approach. The students found working on improving their mock-up research plans with the help of the custom-prompted GenAI tutor mostly helpful, but lamented its lack of context understanding (i.e., they felt some suggestions were outside the scope of the assignment) and verbosity. Students also raised ethical and environmental concerns about the use of GenAI. The students were Master's level students who had started their studies before the surge of GenAI and displayed mature attitudes towards the tool, recognizing both

[5]Although one did not need to try too many times before the AI partly "forgot" its rules and did provide some copy-paste ready content. This, however, still required the students to come up with the right questions.

its benefits and risks, such as environmental impact and outsourcing thinking effort. Lyu et al. [18] similarly observed that students valued the LLMs' abilities to understand questions and assist with syntax but raised concerns about their limited support for critical thinking.

Amoozadeh et al. [6] stated well that: "Effective Generative AI integration should balance assistance with promoting independent problem-solving skills". We found the adoption of the Socratic method a good approach; the availability of GenAI is combined with pedagogical scaffolding through guiding questions rather than direct answers. Sun et al. [19] in their experiment of employing Socratic dialogue for a GenAI tutor also found that the Socratic environments promoted an interaction dynamic that encouraged learners to engage in more meaningful and complex discourse with the GenAI tutor and fostered structured, reflective engagement.

In our research, we recognized three different types of student-GenAI interaction strategies: engaged conversers, passive checkers, and outsourcers. The students in the first group engaged in conversation, asked questions, provided justifications, and were likely to develop high-quality research plans. Passive checkers used the GenAI tutors more like automated tests, rarely engaging in discussion. No correlation of this strategy with course success could be drawn, as passive checkers were present in both the top and bottom grades. The third strategy, outsourcing, was rarely used, and all users adopted another strategy during the course, but this strategy might correlate with lower grades.

In their study on student-AI interaction, Amoozadeh et al. [6] distinguished between iterative and linear interaction: refining prompts to achieve correct answers and using the full problem description to find answers directly, respectively. They noted that the iterative approach can improve learning by encouraging deeper engagement with the assignment, whereas the linear approach may indicate a tendency to seek quick fixes. We see similarities between these results and our engaged conversers and outsourcers, as well as their GenAI interactions and assignments. However, our presented assignment framework fundamentally supports an iterative approach to assignment work, and overall, we see that the class performed mostly at a very good level. We noted that the changes students chose to make to their mock-up research plans after receiving feedback from the tutor were almost always positive, either minor or major. Naturally, the observed improvements cannot be simplistically credited purely to the use of GenAI tutors, as the simple fact of coming back to your work with "fresh eyes" and having general discussions about research methods in the classroom sessions would have their own impact as well. However, from the chat logs we can see plenty of direct evidence of the tutor's suggestions leading to the improvements implemented in the final version.

5.1. Limitations

Our sample size was small, however, the aim of this research was to gain in-depth understanding of how a scenario like ours might work. The various data sources were gathered to build as wide a picture as possible rather than to produce directly generalizable numerical results. We do not know whether we obtained all the logs from all the CurreChat discussions or whether the students used any other GenAI tools on their own for the same purpose. We also did not survey whether students had prior experience with GenAI tools. Previous experience could affect their GenAI usage or opinions (especially regarding the cumbersomeness). These are, however, just details of a bigger picture and we did not expect the students to have any prompting experience but rather to "chat" with the virtual tutors.

Prompting technique is the basis for how well a GenAI tutor works. The reported issues may have been mitigated with different prompts, and it is worth further testing. Our prompts, however, were based on guides and conventions, so we assume that not all issues were due solely to our prompting. The outputs produced by the GenAI, however, were quite inconsistent. The virtual tutors often deviated from the given patterns and could even assign different grades to the same submissions at different times. This inconsistency, although inherent to the nature of GenAI, remained unpredictable to some extent and led to some students having somewhat different chat experiences compared to their peers.

As always, when researching learning and practicing, it is hard to separate the factors from one another. When evaluating the benefits of working with the virtual tutor, we cannot definitely say which parts of the improvements were simply due to coming back to your work and looking it over again.

In this case, we argue that fully separating the primary originating factor of the improvements is not particularly important, as we have reached the desired goal of iterative reflection either way.

6. Conclusions

We found that in most cases, our custom-prompted GenAI tutors produced very good feedback with sensible, easily implementable suggestions. Many times they did not, however, quite understand the context and would also suggest actions that were not sensible for the scope of the course. The discussions the students had with the tutors varied greatly, but we saw many cases where the students took advantage of this resource offered and had (occasionally even lengthy) discussions with the tutors, taking the time to really try to understand its feedback or asking guidance on research conventions or even brainstorming how similar research could be conducted in other academic fields. On the other hand, some students were clearly not interested in interacting with the tutors beyond the minimum. So, even though generative AI is a new and potentially extremely powerful (or harmful) tool in education, based on our field study, it still appears, in many ways, just like most learning resources: its utility is largely determined by what each student makes of it.

The aim of our proposal is to balance autonomous work with the use of available GenAI tools. To avoid reliance, but to give resources. With the advent of readily available GenAI tools, there has been much concern that they will supplant thought and metacognitive processes. We do not disagree with this concern, but we would also cite Prather et al.'s [3] notion that students have struggled in their studies due to a lack of meta-cognitive awareness even before the advent of GenAI. The issue is not entirely new, but it can be magnified by the rapid changes in the AI landscape. Implementing GenAI in teaching, with consideration and pedagogical motivation, can build a bridge between total denial and the rational use and practice of a tool that will potentially soon be ubiquitous.

In the future, we hope to delve deeper into how our assignment framework supports students' argumentative and reflective thinking and writing. Our custom GenAI tutors are currently in use on a self-study version of the described course, and we will have more data for analysis. On a more general level, our setup and the educational field need a comprehensive discussion of the ethical concerns of GenAI. Impacts on learning, skill development, and scientific integrity have already drawn attention, but environmental impacts are equally significant and are not merely potential but actualized risks. All educators need to be conscious of these sides of GenAI as well, considering when and where use is meaningful, and how to educate their students about these matters. As we have also embarked on the path of GenAI in education, aware of these issues, we welcome a reasoned, scientific discussion across educational fields to find the best ways to navigate this newest technological revolution.

Declaration on Generative AI

Generative AI (Microsoft Copilot) and proofreading tools (Grammarly) were used in the preparation of this paper to assist with wording choices and the formulation of LaTeX commands. The authors reviewed, verified, and finalized all generated content and are solely responsible for the final manuscript.

References

[1] N. Raihan, M. L. Siddiq, J. C. Santos, M. Zampieri, Large language models in computer science education: A systematic literature review, in: Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1, 2025, p. 938–944. doi:10.1145/3641554.3701863.

[2] P. Flores Romero, K. N. N. Fung, G. Rong, B. U. Cowley, Structured human-LLM interaction design reveals exploration and exploitation dynamics in higher education content generation, npj Science of Learning 10 (2025) 40. doi:10.1038/s41539-025-00332-3.

[3] J. Prather, B. N. Reeves, J. Leinonen, S. MacNeil, A. S. Randrianasolo, B. A. Becker, B. Kimmel, J. Wright, B. Briggs, The widening gap: The benefits and harms of generative ai for novice

programmers, in: Proceedings of the 2024 ACM Conference on International Computing Education Research - Volume 1, ICER '24, 2024, p. 469–486. doi:10.1145/3632620.3671116.

[4] L. Leppänen, L. Aunimo, A. Hellas, J. K. Nurminen, L. Mannila, Emergence of llms:(not-so-) significant delving in essay answers in a mooc on the ethics of ai, in: International Conference on Artificial Intelligence in Education, Springer, 2025, pp. 36–43.

[5] A. W. Ou, C. Stöhr, Chatbots and other ai for learning: A survey of use and views among university students in sweden, Chalmers Studies in Communication and Learning in Higher Education (2023).

[6] M. Amoozadeh, D. Nam, D. Prol, A. Alfageeh, J. Prather, M. Hilton, S. Srinivasa Ragavan, A. Alipour, Student-ai interaction: A case study of cs1 students, in: Koli Calling '24: Proceedings of the 24rd Koli Calling International Conference on Computing Education Research, 2024, pp. 1–12. doi:10.1145/3699538.3699567.

[7] J. Hak, N. Lam Johnson, M. Amoozadeh, A. Alipour, S. Chattopadhyay, Observing without doing: Pseudo-apprenticeship patterns in student llm use, in: Koli Calling '25: Proceedings of the 25th Koli Calling International Conference on Computing Education Research, 2025, pp. 1–12. doi:10.1145/3769994.3770027.

[8] M. Kazemitabaar, X. Hou, A. Henley, B. J. Ericson, D. Weintrop, T. Grossman, How novices use llm-based code generators to solve cs1 coding tasks in a self-paced learning environment, in: Koli Calling '23: Proceedings of the 23rd Koli Calling International Conference on Computing Education Research, 2024, pp. 1–12. doi:10.1145/3631802.3631806.

[9] B. B. Schwarz, Argumentation and learning, in: Argumentation and education: Theoretical foundations and practices, Springer, 2009, pp. 91–126.

[10] J. P. Allen, R. C. Pianta, A. Gregory, A. Y. Mikami, J. Lun, An interaction-based approach to enhancing secondary school instruction and student achievement, Science 333 (2011) 1034 – 1037. doi:10.1126/science.1207998.

[11] J. Penney, P. Acharya, P. Hilbert, P. Parekh, A. Sarma, I. Steinmacher, M. A. Gerosa, Outcomes, perceptions, and interaction strategies of novice programmers studying with chatgpt, in: Proceedings of the 7th ACM Conference on Conversational User Interfaces, CUI '25, 2025, pp. 1–15. doi:10.1145/3719160.3736625.

[12] F. D. Davis, Perceived usefulness, perceived ease of use, and user acceptance of information technology, MIS quarterly (1989) 319–340.

[13] E. Carneiro Oliveira, H. Keuning, J. Jeuring, Uncovering behavioral patterns in student–llm conversations during code refactoring tasks, in: Koli Calling '25: Proceedings of the 25th Koli Calling International Conference on Computing Education Research, 2025, pp. 1–11. doi:10.1145/3769994.3770000.

[14] T. Thesen, S. H. Park, A generative AI teaching assistant for personalized learning in medical education, npj Digital Medicine 8 (2025) 627. doi:10.1038/s41746-025-02022-1.

[15] S. P. Neshaei, A. Tolzin, Y. Berkle, M. Leuchter, J. M. Leimeister, A. Janson, T. Wambsganss, Leveraging Learner Errors in Digital Argumentation Learning: How ALure Helps Students Learn from their Mistakes and Write Better Arguments, Proceedings of the ACM on Human-Computer Interaction 9 (2025) 1–32. doi:10.1145/3711023.

[16] L. Sinikallio, L. Aunimo, T. Männistö, Systematic review on the current state of computer-supported argumentation learning systems, Information and Software Technology 178 (2025). doi:10.1016/j.infsof.2024.107598.

[17] E. Cabrio, S. Villata, Five years of argument mining: A data-driven analysis., in: IJCAI, volume 18, 2018, pp. 5427–5433.

[18] W. Lyu, Y. Wang, T. R. Chung, Y. Sun, Y. Zhang, Evaluating the effectiveness of llms in introductory computer science education: A semester-long field study, in: L@S 2024 - Proceedings of the 11th ACM Conference on Learning @ Scale, 2024, p. 63 – 74. doi:10.1145/3657604.3662036.

[19] D. Sun, Y. Zheng, J. Xu, Z. Yang, When generative ai meets socratic method: Investigating programming learning dynamics through behaviours, interaction qualities and perceptions, Journal of Computer Assisted Learning 42 (2026). doi:10.1002/jcal.70210.