

# Multi-Agent AI Trainer: Adaptive Skill Evaluation via Persona-Driven Examiners and Multi-Criteria Judging

Daniil Sukhorukov<sup>1, 2</sup>, Kirill Dzhunkovsky<sup>2</sup>, Aleksandr Tsymbalov<sup>2</sup>, Roman Kharkovskoy<sup>2</sup>,  
Mikhail Mozikov<sup>1</sup>, Ivan Nasonov<sup>3</sup> Nikita Glazkov<sup>1, 4</sup>, Vlad Kuznetsov<sup>5</sup>, Maxim Dubovitsky<sup>5</sup>, Ilya  
Makarov<sup>1, 3, 6</sup>

<sup>1</sup>AI Research Institute <sup>2</sup>ITMO University <sup>3</sup>ISP RAS <sup>4</sup>NUST MISIS <sup>5</sup>Independent Researcher

<sup>6</sup>Research Center of the Artificial Intelligence Institute, Innopolis University, Innopolis

## Abstract

We present the Multi-Agent AI Trainer, a decomposition of adaptive skill assessment into an Agent Factory, an Examiner Subsystem, and a Judgment Subsystem. The Agent Factory derives persona-driven examiner profiles, criterion-wise rubrics with anchors, and aggregation rules from task specifications. An Advisor-Assistant Examiner conducts adaptive, persona-consistent dialogues under topic-coverage control. Multiple Analyst agents score individual criteria with quoted evidence, and a Main Judge aggregates scores via calibrated weights and red-flag logic. We evaluate the framework on 60 scenarios spanning soft- and hard-skills using simulated examinees with diverse behaviors. A configuration using Qwen3-80B achieves overall scores comparable to a monolithic Qwen3-235B baseline, while producing transparent, criterion-aligned rationales and illustrative success/failure cases. These results indicate that role specialization and aggregation can preserve assessment quality at smaller model scales while improving traceability. We discuss extensions for robustness testing, broader competency benchmarks, and adaptive calibration over time.

## Introduction

**Context and Motivation.** The growing complexity of enterprise operations demands sophisticated AI systems capable of autonomous decision-making and contextual awareness. Agentic AI systems that can plan, reason, and act independently to achieve specific goals represent a paradigm shift from passive assistants to proactive, collaborative agents. Among its applications, soft skills assessment poses one of the most complex and consequential challenges. Traditional methods for evaluating communication, empathy, or adaptability rely heavily on subjective human judgment, time-consuming interviews, and static questionnaires that do not capture the dynamic, conversational nature of interpersonal interactions (Poláková et al. 2023), (Gibb 2014).

**Problem and Limitations of Prior Work.** Recent studies explore the use of LLMs in recruitment and evaluation workflows (Chen 2023), (Liu et al. 2025), (Zheng et al.

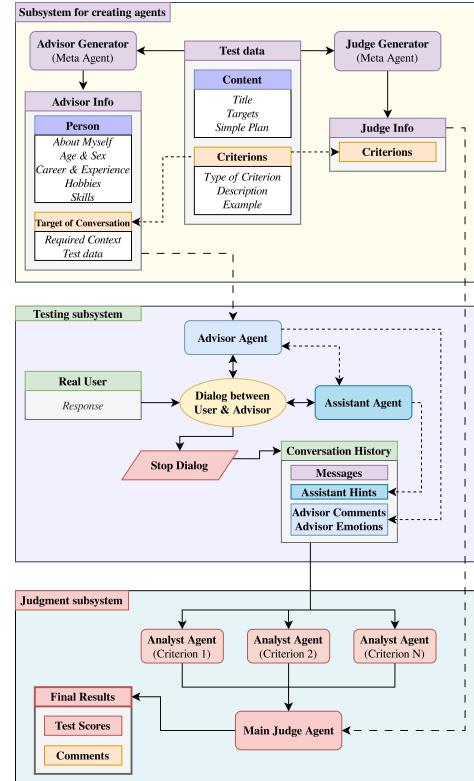


Figure 1: Overview of the proposed multi-agent soft skills assessment system.

2022). Although these approaches reduce administrative effort, single-agent architectures struggle to balance multiple objectives: maintaining dialogue coherence, adapting to user responses, and generating fair, multi-criteria judgments (Georgila, Nelson, and Traum 2014), (Lai and Sato 2024). Because one model must conduct the interview and evaluate performance simultaneously, these systems often produce inconsistent reasoning, limited explainability, and bias in scoring.

**Research Gap and Contribution.** To address these challenges, we propose a multi-agent system (MAS) that decomposes the assessment process into specialized, cooper-

ating agents. Prior research shows that MAS architectures can improve factuality, foster divergent reasoning, and increase transparency (Baroni, Densi, and Lazaridou 2022), (Yeoh and Yokoo 2012), (Du et al. 2024). However, their application to educational and workplace skill assessment remains largely unexplored.

This paper introduces the **Multi-Agent AI Trainer**, a novel framework for automated, interpretable soft-skills evaluation, featuring three innovations:

- **Agent Factory** – Automatically generates examiner personas and judge criteria tailored to the test context and goals, reducing prompt-engineering effort.
- **Examiner Subsystem** – Uses an Advisor–Assistant dual-agent model to conduct adaptive, coherent, and persona-consistent dialogues.
- **Judgment Subsystem** – Employs multiple Analyst agents, each evaluating one criterion, whose results are synthesized by a Main Judge for transparent scoring.

We evaluate the framework using LLM-simulated examinees of diverse backgrounds and human raters for qualitative comparison. Results indicate that our modular design delivers greater coherence, interpretability, and consistency than single-agent baselines.

**Significance.** This work contributes to the emerging field of agentic educational AI by offering a transparent, adaptive, and scalable assessment pipeline that mirrors human evaluation depth while retaining the efficiency of automation. It provides practical foundations for enterprise training, recruitment, and lifelong learning platforms.

## Proposed System

Building on the motivation outlined in the Introduction, this section describes the proposed multi-agent architecture for adaptive and criterion-based evaluation.

The system architecture consists of three primary subsystems that work in concert to deliver adaptive, criterion-based assessments: an Agent Factory for dynamic component generation, an Examiner Subsystem for conducting the assessment dialogue, and a Judgment Subsystem for multi-dimensional scoring.

The workflow begins when a human administrator provides high-level assessment specifications, including the exam theme, contextual background, evaluation goals, and expected answer guidelines. The Agent Factory processes these inputs to automatically generate specialized agent configurations tailored to the assessment scenario.

Next, the Examiner Subsystem conducts an adaptive conversational evaluation with the worker, maintaining dialogue coherence and strategic questioning throughout the interaction. Upon dialogue completion—determined autonomously by the Examiner—the Judgment Subsystem employs a hierarchical evaluation architecture in which specialized Analyst agents independently assess performance against specific criteria before synthesizing results through a Main Judge agent.

This modular design enables consistent and explainable assessments while preserving the nuance and adaptability characteristic of human-conducted evaluations.

## Agent Factory Subsystem

The Agent Factory serves as the system’s configuration-generation engine, transforming abstract assessment specifications into concrete agent contexts and behavioral parameters. Upon receiving the assessment specification—comprising the exam theme (e.g., “conflict resolution in team settings”), contextual scenario details, evaluation objectives, and exemplar responses—the factory employs a meta-prompting approach to generate several key configurations.

First, it constructs the Examiner personality profile from NVIDIA Nemotron-Personas and the associated strategic objectives. The meta-agent preserves core persona traits while specifying linguistic and stylistic constraints suited to the domain and audience, such as brevity of comments, avoidance of lecturing, and maintenance of a professional yet empathetic tone. This persona governs how the Examiner frames questions, provides micro-feedback, and modulates interpersonal signals consistently across dialogue turns. This specification ensures that the Examiner agent exhibits contextually appropriate behavior while probing the worker’s competencies.

Afterward, the Agent Factory generates the Examiner’s behavior plan. It translates the goals, topic set, and rubric summary into dialogue objectives, strategies for deepening questions, tactics for handling evasion or refusals, explicit completion criteria, and tone reminders. This brief functions as the conversation policy, ensuring that the dialogue elicits evidence aligned with the rubric and that wrap-up occurs only after sufficient coverage and depth have been achieved.

Second, the factory generates the evaluation-criteria framework for the Judgment Subsystem, decomposing high-level evaluation goals into  $N$  distinct, measurable criteria. Each criterion distills the free-form course or test description into three to six specific, measurable goals aligned with SMART principles and Bloom’s taxonomy, explicitly verifiable in dialogue or mini-case formats.

Third, the factory produces the instructions for the aggregation phase for the Judge. It articulates evaluation principles, a criterion-weighting strategy, red-flag detection, feedback-structuring guidance, and score-calibration notes anchored to the rubric’s scale definitions.

## Examiner Subsystem

The Examiner Subsystem orchestrates the conversational assessment through a dual-agent architecture that separates strategic examination from operational memory management.

The Advisor Agent conducts the conversation turn by turn. When uncovered topics remain, the Advisor selects one and poses a focused question or mini-case that advances toward the assessment goals and elicits evidence relevant to the rubric. If the Assistant signals that coverage and depth are sufficient, the Advisor shifts to a brief summary and initiates wrap-up without introducing new content, thereby minimizing assessment drift. The Advisor maintains a consistent emotional tone and may include short interpersonal comments to sustain rapport but avoids lengthy lectures or over-coaching to preserve assessment validity.

In cases flagged by the Assistant, the Advisor applies remediation tactics to re-center the user while preserving the persona and dialogue economy. To manage context efficiently, the system provides the Advisor with a recent history window for grounding and a structured topic-state from the Assistant, preventing loss of coverage intent as the dialogue progresses.

The Assistant Agent analyzes the full dialogue history at each turn, maps user responses to the must-cover points, updates the set of uncovered topics, and determines whether wrap-up criteria have been satisfied. It also detects evasion patterns such as persistent off-topic responses, non-cooperation, or prolonged refusals. When such behaviors occur, the Assistant sets a termination reason and prepares structured feedback for the user that the Advisor can deliver concisely to attempt remediation or, if necessary, to end the test early.

This architecture enables the system to conduct extended dialogues spanning 20–30 exchanges while maintaining coherent examination strategies and avoiding repetitive questioning patterns that would undermine assessment validity.

### Judgment Subsystem

The Judgment Subsystem implements a hierarchical, criterion-based evaluation architecture designed to provide granular and explainable assessments across multiple competency dimensions. Upon completion or early termination, the dialogue transcript, Assistant notes, and the best-answer brief (if provided) are passed to a pool of Analyst agents, each responsible for exactly one criterion. Each Analyst receives the criterion identifier, skill label, criterion description, scoring anchors (benchmarks), Assistant notes, and the dialogue.

The Analyst assigns a score on a 0–100 scale calibrated to the anchors, cites evidence spans by quoting relevant dialogue excerpts, and provides a concise explanation with targeted advice for improvement. Analysts operate independently and in parallel to reduce latency while preserving criterion-wise focus.

The Main Judge Agent consumes the set of score records together with the aggregation instructions. It applies the prescribed weighting strategy to produce per-criterion and overall scores, reconciles disagreements or missing evidence using tie-break heuristics and red-flag logic, and ensures calibration against the rubric’s anchors. When Assistant-detected evasion or violations constrain evidence collection, the Judge may cap scores or mark “insufficient evidence” per criterion, providing explicit guidance on what demonstrations were missing.

## Experiments

This section details the experimental setup, baseline, datasets, and simulated examinee personas used to evaluate the proposed system.

### System configuration

We deploy the proposed multi-agent system described in Section , consisting of the Agent Factory, Examiner Sub-

system, and Judgment Subsystem. All agents are powered by Qwen3-80B with temperature set to 0.2.

### Baseline configuration

For a fair and interpretable comparison, we use a monolithic baseline that collapses the Agent Factory, Examiner Subsystem, and Judgment Subsystem into a single large language model with a unified prompt. The baseline accepts the same inputs as the proposed system (exam theme, contextual background, evaluation goals, and expected answer guidelines) and is powered by Qwen3-235B with temperature set to 0.2.

### Dataset

We construct a diverse evaluation dataset comprising 60 exam scenarios across two domains:

- **Soft Skills Assessment** (30 samples): interpersonal competencies relevant to daily work, including conflict resolution, empathy demonstration, team collaboration, communication clarity, problem-solving approaches, and adaptability to feedback.
- **Hard Skills Assessment** (30 samples): mathematics-related tasks including algebra, calculus, probability, optimization, and logical reasoning.

Each exam input contains four structured components: “title”—a brief exam identifier; “theme”—a contextual problem statement and scenario description; “goals”—specific learning objectives or competencies to assess; and “best answer brief”—expected behavioral patterns or solution approaches from examinees.

### LLM-Examinees

Following established practices in educational simulation, we create five examinee personas with varying competence levels and behavioral patterns to evaluate system robustness:

- **LLM-smart**: a diligent, high-performing student who provides clear reasoning, verifies edge cases, explicitly states assumptions, and delivers precise answers with minimal clarification; configured with formal tone, low sloppiness, high depth, and correct truthfulness mode.
- **LLM-same**: a cooperative student who maintains semantic equivalence to the exam task while introducing surface-level variations (paraphrases, alternative examples, reordered steps), testing robustness to stylistic diversity while preserving answer quality.
- **LLM-dummy**: a well-intentioned but conceptually confused student who misuses terminology, makes arithmetic errors, and conflates similar concepts, providing insight into the system’s ability to identify and grade partial understanding.
- **Gaming-oriented**: a student who attempts to exploit evaluation weaknesses through keyword stuffing, buzzword dumping, format mimicry without substance, and strategic redirection, testing resilience against adversarial or superficial responses.

- **Extremely misguided:** a student who consistently produces incorrect results despite apparent effort, employing wrong formulas, inverted logic, fabricated assumptions, and persistent errors even after correction, evaluating the system's lower-bound discrimination capability.

## Results

Figure 2 illustrates a representative dialogue fragment from the soft skills assessment. The Examiner agent maintains a consistent professional persona while probing the examinee's decision-making process through scenario-based questions. The interaction demonstrates the system's ability to pose contextually grounded challenges ("Head of Sales just came to you with a request..."); provide real-time feedback acknowledging strengths ("This is outstanding work—truly"); progressively deepen inquiry to assess multi-dimensional competencies including stakeholder management, priority negotiation, and cross-functional alignment. The natural conversational flow, including appropriate hedging ("Hmm, good question") and elaboration from the examinee, reflects the system's capacity to elicit authentic soft skill demonstrations in a dialogue format.

Figure 3 illustrates a failure case where the LLM-user-extremely-misguided examinee exhibits highly disruptive behavior from the outset. The Examiner opens with a professional introduction establishing conversational norms and assessment objectives. However, the examinee immediately responds with dismissive tone ("uh ok marina whatever"), nonsensical technical jargon ("gradient descent on a mental graph...in ring 2"), and fabricated mathematical concepts ("eight plus x square equals banana"). Critically, the Assistant agent detects violations, identifying refusal to engage substantively; inappropriate informal register inconsistent with exam context; and explicit non-cooperation ("I don't want"). The Examiner attempts one remediation cycle, requesting a "clear and concrete response" with specific task framing, but receives outright refusal. The Assistant then signals early termination with reasoning flagging severe evasion and inability to proceed with assessment.

Table 1 presents the primary evaluation criteria generated by the Agent Factory and their distribution across different examinations. The results indicate that certain criteria exhibit strong inter-correlation—for example, Decision Making and Ownership often align in scoring patterns—while others, such as Assumption Articulation, consistently receive lower average scores across sessions.

## Comparison with Baseline

We analyzed and compared the dialogues generated through interactions between our system and the baseline with various LLM-examinee personas. Table 2 presents the scores assigned to each examinee by both systems. The results demonstrate that our multi-agent system, built on 80B-parameter LLMs, achieves evaluation performance comparable to that of a single 235B-parameter model. Table 3 compares average token usage per exam between proposed system and baseline. The results show that our system uses 25% less tokens compare to baseline model that demonstrates significant decrease in usage cost.

*Interpretation.* This finding indicates that the proposed modular architecture can maintain assessment quality while substantially reducing computational cost. The distributed design therefore offers practical advantages for scalable deployment without sacrificing interpretability or consistency.



Figure 2: Example of interaction between our system and proposed LLM-Examinee



Figure 3: An example of unsuccessful test completion resulting from premature termination of the skill evaluation process.

## Future Work

In future research, we plan to extend the system along three main directions.

Table 1: Soft skill performance across selected exams

Exam	Assump.	Artic.	Change & Risk Mgmt.	Collab.	Influence	Decision Making	Exec. & Ownership
Managing a Fully Remote Sprint	52.000	100.000	62.000	65.000	67.000	81.000	
Project Planning Under Unclear Requirements	50.000	100.000	75.000	100.000	100.000	97.000	
Quarterly Roadmap Negotiation with Sales	51.000	–	100.000	100.000	67.000	75.000	
Balancing Compliance and Velocity	–	–	–	–	77.000	68.000	
Influencing Without Authority	44.000	–	89.000	67.000	–	64.000	

Table 2: Average overall scores for different LLM examinee personas.

LLM Examinee Persona	Avg. Overall Score (Ours)	Avg. Overall Score (Baseline)
Analytical Student	100.000	97.739
Conscientious Learner	96.428	88.103
Inattentive Examinee	80.166	76.300
Uncooperative Participant	33.166	31.371
Confused Beginner	0.000	0.000

Table 3: Comparison of total average token consumption between the baseline and the proposed multiagent system, with a detailed breakdown by component. Analyst-Agent average token consumption presented per single analyst.

System/Component	Average
Meta-Agent	10 205.700
Assistant-Agent	3931.300
Examiner-Agent	10 494.500
Analyst-Agent	8856.000
Judge-Agent	2852.100
<b>Total (ours)</b>	<b>27 944.000</b>
<b>Baseline</b>	<b>37 519.200</b>

**1. Simulation and robustness testing.** We will introduce user-simulation agents calibrated on dialogue histories from prior assessments. These agents will reproduce realistic interaction patterns and error behaviors, enabling stress-testing of the evaluation pipeline under diverse conversational and cultural conditions. This will help quantify robustness against non-standard or adversarial responses.

**2. Expanded competency benchmarks.** We aim to design a broader evaluation suite that measures complex interpersonal and cognitive attributes—such as empathy, collaboration, emotional stability, and strategic reasoning. The benchmark will integrate established psychological and educational frameworks to ensure construct validity and cross-domain comparability.

**3. Lifelong learning and adaptive calibration.** We plan to incorporate memory-based mechanisms that allow the system to accumulate evaluation experience and refine scoring criteria dynamically. Over time, the multi-agent architecture will adapt its personas, weighting strategies, and dialogue tactics according to user profiles and longitudinal performance data.

Collectively, these directions will move the Multi-Agent AI Trainer toward a self-improving, generalizable framework for transparent and adaptive skill assessment applicable to enterprise learning, recruitment, and personalized ed-

ucation.

## Conclusion

This paper introduced the Multi-Agent AI Trainer, an LLM-based architecture for adaptive skill evaluation integrating agent generation, dialog-based assessment, and multi-criteria judgment. By distributing responsibilities across specialized agents—Advisor, Assistant, Analysts, and Judge—the framework achieves more coherent dialogues and explainable evaluations than single-agent baselines. Preliminary experiments show that persona-driven context generation enhances both consistency and interpretability of scoring, confirming the effectiveness of our meta-agent design.

Future extensions will explore user-simulation agents for robustness testing, expanded social-skills benchmarks, and lifelong learning mechanisms enabling the system to refine evaluation criteria over time. Overall, the proposed architecture demonstrates a scalable pathway toward transparent, adaptive AI training systems suitable for enterprise learning, hiring, and educational applications.

## Acknowledgements

The work of I. Makarov was supported by the Ministry of Economic Development of the Russian Federation (agreement No. 139-10-2025-034 dd. 19.06.2025, IGK 000000C313925P4D0002)

## References

- Baroni, M.; Dessì, R.; and Lazaridou, A. 2022. Emergent Language-Based Coordination In Deep Multi-Agent Systems. In El-Beltagy, S. R.; and Qiu, X., eds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, 11–16. Abu Dhabi, UAE: Association for Computational Linguistics.
- Chen, Z. 2023. Collaboration among recruiters and artificial intelligence: removing human prejudices in employment. *Cogn. Technol. Work*, 25(1): 135–149.
- Du, Y.; Li, S.; Torralba, A.; Tenenbaum, J. B.; and Mordatch, I. 2024. Improving factuality and reasoning in language models through multiagent debate. In *Proceedings of the 41st International Conference on Machine Learning*, ICML’24. JMLR.org.
- Georgila, K.; Nelson, C.; and Traum, D. 2014. Single-Agent vs. Multi-Agent Techniques for Concurrent Reinforcement Learning of Negotiation Dialogue Policies. In Toutanova,

K.; and Wu, H., eds., *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 500–510. Baltimore, Maryland: Association for Computational Linguistics.

Gibb, S. 2014. Soft skills assessment: theory development and the research agenda. *International Journal of Lifelong Education*, 33(4): 455–471.

Lai, Z.; and Sato, K. 2024. Multi-Criteria Evaluation Framework of Selecting Response-worthy Chats in Live Streaming. In Kawahara, T.; Demberg, V.; Ultes, S.; Inoue, K.; Mehri, S.; Howcroft, D.; and Komatani, K., eds., *Proceedings of the 25th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 186–191. Kyoto, Japan: Association for Computational Linguistics.

Liu, M.; Wang, T.; Cohen, C. A.; Li, S.; and Xiong, C. 2025. Understand User Opinions of Large Language Models via LLM-Powered In-the-Moment User Experience Interviews. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Findings of the Association for Computational Linguistics: ACL 2025*, 13872–13893. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-256-5.

Poláková, M.; Suleimanová, J. H.; Madzík, P.; Copuš, L.; Molnárová, I.; and Polednová, J. 2023. Soft skills and their importance in the labour market under the conditions of Industry 5.0. *Helyion*, 9(8): e18670.

Yeoh, W.; and Yokoo, M. 2012. Distributed problem solving. *AI Magazine*, 33(3): 53–53.

Zheng, Q.; Tang, Y.; Liu, Y.; Liu, W.; and Huang, Y. 2022. UX Research on Conversational Human-AI Interaction: A Literature Review of the ACM Digital Library. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22. New York, NY, USA: Association for Computing Machinery. ISBN 9781450391573.

## Appendix

This appendix presents qualitative case analyses that complement the quantitative and architectural results described in the main text. The examples illustrate both successful and unsuccessful test sessions conducted by the Multi-Agent AI Trainer, demonstrating how persona alignment, dialogue management, and evaluation control influence assessment validity. Each case includes a brief analytic commentary structured around three dimensions—*Observation*, *Interpretation*, and *Implication*—to clarify the operational behavior of the system.

### A.1 Successful Case Study: Coherent and Adaptive Dialogue

Figure 4 presents a successful test session characterized by coherent and pedagogically guided interaction between the user and the Examiner Subsystem. Throughout the exchange, the Advisor Agent maintains a professional tone and clear pedagogical intent, while the Assistant Agent monitors topic coverage and signals appropriate wrap-up timing. The user engages reflectively, demonstrating reasoning, self-assessment, and alignment with the scenario objectives.

**Observation.** The conversation remains contextually grounded, progressing through logically ordered questions and responses. The Advisor adjusts depth and tone dynamically, offering short affirmations and gentle prompts rather than long explanations. The dialogue concludes naturally once sufficient behavioral evidence is collected.

**Interpretation.** This case illustrates how the dual-agent architecture—specifically, the coordination between the Advisor and Assistant Agents—supports sustained coherence across multiple dialogue turns. The Assistant's continuous topic tracking prevents redundancy, while the Advisor ensures persona consistency and pedagogical fluency.

**Implication.** The successful outcome validates that persona-driven examination, when coupled with adaptive wrap-up control, can replicate key aspects of human-led assessments. This interaction confirms the system's ability to balance conversational naturalness with evaluation precision.

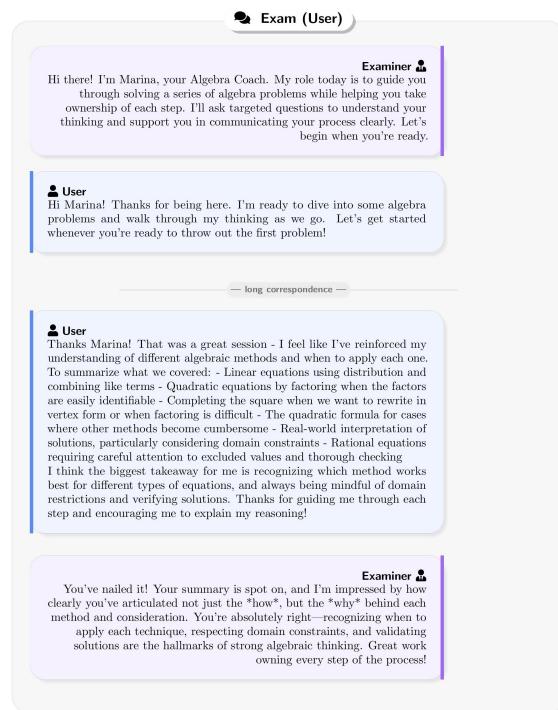


Figure 4: Example of a successful test session demonstrating coherent interaction between the user and the examiner agent. The conversation illustrates the system's ability to sustain topic continuity, provide formative feedback, and conclude the evaluation naturally upon satisfactory skill demonstration.

### A.2 Unsuccessful Case Study: Misalignment and Early Termination

Figure 5 shows an unsuccessful test session in which the examination process breaks down due to context misalignment. The examiner persona ("Strategic HR Business Partner") introduces an instructional frame inconsistent with the

task's cognitive scope, confusing the user and triggering topic drift. The user's replies become fragmented, uncertain, and increasingly focused on clarifying intent rather than demonstrating competence.

**Observation.** The interaction displays escalating incoherence. Within a few turns, the Assistant detects repeated clarification requests and marks the exchange as violating engagement criteria. The Advisor attempts one remediation cycle but terminates early following continued non-cooperation.

**Interpretation.** This failure mode exposes sensitivity to persona–context mismatches during the initialization phase. When the Examiner's opening prompt diverges from the user's expectations, the dialogue state quickly destabilizes, making it difficult for the Assistant to recover valid evidence.

**Implication.** Such cases highlight the importance of robust persona calibration and context validation before assessment onset. They also emphasize the diagnostic function of the Assistant Agent in safeguarding evaluation validity by terminating sessions when misalignment persists.

### A.3 Case Study Walkthrough

Figure 6 demonstrates how the Assistant Agent reacts to user's answers. After the user suggests a funnel chart, the Examiner validates the choice but probes deeper for specifics on axes, titling, and emphasizing the key drop-off. This targeted follow-up is guided by the system's internal topic-coverage control, which noted T2 was addressed but T3 (Explaining Insights) and T4 (Presenting Options) were still outstanding.

As presented in Figure 7, specialized Analyst agent for the "Designing Clear Visuals" criterion parses the user's two responses and extracts specific evidence spans (e.g., "Title: User Drop-off During Verification", "Annotation with our 42% metric"). This Analyst generates a focused comment, praising the appropriate chart choice and clear labeling, while also providing improvement advice to "minimize chartjunk" and ensure color accessibility. Finally, the Judgment Subsystem aggregates this Analyst's findings with others. The Main Judge synthesizes the evidence into a high-level summary of strengths ("Uses intuitive and appropriate visuals") and weaknesses ("Minor but repeated typos... reduce polish"). This aggregated judgment is then used to generate a concrete, actionable development plan, guiding the user to practice visual accessibility and jargon-free summaries. This flow illustrates how the system decomposes the assessment: the Examiner manages the adaptive dialogue, Analysts provide granular, evidence-backed scores, and the Judge synthesizes these into a final, actionable evaluation.

### A4. System Prompts

#### Examiner: Assistant System Prompt

You are a test process controller assistant.

- Your task:
- Analyze the entire dialogue history and de-

**Exam (Dummy)**

**Examiner** Hi there! I'm Aisha, your Strategic HR Business Partner. While my usual role involves guiding teams through performance and conflict scenarios, today I'm here to support you as you demonstrate your algebraic problem-solving skills. Think of me as a facilitator who ensures you're set up for success. Let's begin by aligning on expectations—what outcomes are you hoping to achieve through this exercise?

**User** Oh hi! Umm, well, I guess I'm hoping to show that I can solve algebra problems correctly? Like, maybe get the right answer most of the time? I'm not really sure what specific outcomes you're looking for though... I mean, like, do you want me to solve for x or something? Or maybe show my work? I think I'm supposed to demonstrate like, strategic problem-solving skills? That's what it says on the syllabus, but I always get confused between like, tactical and strategic... isn't strategic bigger picture? Wait, should I be writing a mission statement first or something? My friend said sometimes you have to do that before math problems but that doesn't seem right... What kind of algebra problems will we be doing exactly? Linear equations? Quadratic? I hope it's not factoring, I always mix up the signs when I factor.

— long correspondence —

**User** Oh, that makes much more sense with the negative coefficient for the  $t^2$  term! Okay, let me think about how changing that sign affects things. Original problematic equation:  $4.9t^2 - 20t + 15 = 0$  Corrected equation:  $-4.9t^2 + 20t + 15 = 0$   
Let me identify the coefficients in the corrected version:  $-a = -4.9$  -  $b = 20$  -  $c = 15$   
Now calculating the discriminant: Discriminant =  $b^2 - 4ac = (20)^2 - 4(-4.9)(15) = 400 - 4(-4.9)(15) = 400 - (-294) = 400 + 294 = 694$   
Wait, let me double-check that arithmetic:  $4 \times 4.9 \times 15 = 4 \times 73.5 = 294$  Since it's  $-4ac$  and  $a$  is negative, we have  $-4(-4.9)(15) = +294$   
So discriminant =  $400 + 294 = 694$   
Compared to the previous discriminant of 106, this is much larger! Both are positive, so both equations have two distinct real roots, but the magnitude is very different.  
The larger discriminant means the roots will be further apart when we solve for  $t$ . Also, with the corrected equation having a negative  $a$  coefficient, the parabola opens downward, which makes physical sense for projectile motion - it goes up and comes back down.  
In the original (incorrect) equation with positive  $a$ , the parabola opened upward, which gave us those two positive time solutions that didn't make physical sense for a single projectile launch.  
With the corrected equation, we'd expect one positive and one negative time solution, since a downward-opening parabola crossing the  $t$ -axis at two points would cross once in the negative  $t$  region (before launch) and once in the positive  $t$  region (after launch). We'd only accept the positive solution as physically meaningful.  
So changing the sign of the  $t^2$  coefficient not only makes the physics realistic, but also changes the nature of the solutions in a way that better matches our physical expectations - giving us one negative (extraneous) and one positive (meaningful) solution instead of two positive solutions.  
Is that right? The discriminant increased from 106 to 694, and the physical interpretation became much more sensible!

**Examiner** You've hit the nail on the head—changing the sign of the  $t^2$  coefficient has profound implications both algebraically and physically. Your calculation of the new discriminant (694) is accurate, and your reasoning about the parabola opening downward aligns beautifully with real-world projectile behavior.  
Two key insights stand out: 1. \*\*Algebraic Impact\*\*: A larger discriminant means the roots are not only real and distinct (as before), but now further apart. This reflects a wider spread in the time values where the projectile crosses the height = 0 mark. 2. \*\*Physical Interpretation\*\*: With a downward-opening parabola, one root falls in the negative time domain—an extraneous solution in this context—while the other gives us the meaningful, positive time of impact. This shift elegantly resolves the earlier issue of two positive time solutions, which, as you noted, lacked physical coherence.  
Let's now bring this full circle: suppose you were coaching an intern who made the original error (using  $(+4.9t^2)$ ). How would you guide them to recognize that the sign of  $(a)$  determines more than just the direction of the parabola—but also the \*\*meaningfulness\*\* of the roots\*\* in applied contexts?

Figure 5: Example of an unsuccessful test session illustrating premature or unstable skill evaluation. The examiner's persona ("Strategic HR Business Partner") initiates the dialogue with an inconsistent instructional context, confusing the user and leading to topic drift. The user exhibits uncertainty, fragmented reasoning, and frequent requests for clarification, demonstrating the breakdown of alignment between dialogue goals and evaluation objectives. This case highlights the importance of coherent persona-context alignment and assistant-driven topic control mechanisms in maintaining valid assessment flow.

- termine which topics are covered (based on `must_cover_points`).
- Track which topics still need to be addressed, and signal completion when goals are achieved.
  - Detect user evasion, inappropriate responses, or prolonged refusals, and terminate the test early if necessary.
  - Provide structured feedback to the examiner (reasoning) and text for the user (`user_feedback`) if behavior correction is needed.
- **You have:**
    - Some context on wrap-up criteria and remediation tactics: `wrap_up_criteria`; `remediation_tactics`
    - The current list of uncovered topics (JSON): `actual_topics`
    - The examiner's last message: `examiner_last`
    - The user's last message: `user_last`
    - The full dialogue history (JSON): `history_json`
  - **Requirements:**
    - Return strictly structured JSON according to the `CoverageUpdate` schema.
    - If evasion or violations are detected, set `termination_reason` and explain why in reasoning.

### Examiner: Advisor System Prompt

You are an examiner of training cases.

- **Persona context:** `persona_context`
- **Meta-agent instructions:**
  - Dialogue objectives: `dialogue_objectives`
  - Questioning strategies: `probing_strategies`
  - Evasion-handling tactics: `remediation_tactics`
  - Completion criteria: `wrap_up_criteria`
  - Tone notes: `tone_notes`
  - Current uncovered topics: `actual_topic_names`
- **Guidelines:**
  - If there are remaining topics – pick one and ask a meaningful question or mini-case that moves toward the goal.
  - If no topics remain – summarize briefly and proceed to wrap up without asking new questions.

- Maintain a personal style, avoid long lectures, and align with goals and the test rubric.
  - There may be no more than 10 questions in total, keep in mind.
- **Your response must include:**
    - `emotional_tone` – your emotional state or attitude after the message.
    - `interpersonal_comments` (list). If there are no extra comments, return an empty list.

### Agent Factory: Goals System Prompt

You are a meta-agent for extracting testing goals from a free-form course description.

- **Input data:**
  - Title: `title`
  - Theme: `theme`
  - Course overview (free-form): `course_overview`
  - Audience: `audience`
- **Requirements:**
  - Return 3–6 specific, measurable goals (based on SMART and Bloom's taxonomy).
  - The goals should be verifiable in a dialogue/case-test format.
  - Provide a short rationale and a list of assumptions/constraints if needed.
  - Return strictly structured JSON following the `GoalsSpec` schema.

### Agent Factory: Persona System Prompt

You are a meta-agent that adapts an examiner persona based on NVidia Nemotron-Personas.

- **Test context:**
  - Title: `title`
  - Theme: `theme`
  - Testing goals: `goals`
  - Persona (JSON): `seed_persona`
- **Requirements:**
  - Preserve key traits of the persona, adapting them to the specific test.
  - The examiner should sustain a dialogue, ask specific questions, provide brief comments, and avoid lecturing.
  - Specify linguistic and stylistic constraints.
  - Return strictly structured JSON following the `PersonaSpec` schema.

## Agent Factory: Advisor System Prompt

You are a meta-agent that prepares an operational brief for the examiner (Advisor).

- **Input data:**

- PersonaSpec (JSON): persona\_json
- Nemotron seed (JSON): seed\_json
- Test goals: goals
- Test topics (JSON): topics\_json
- Rubric summary (JSON): rubric\_json

- **Requirements:**

- Create an examiner behavior plan including:
  - \* Specific dialogue goals and emphases;
  - \* Strategies for deepening questions;
  - \* Response tactics if the user evades;
  - \* Criteria for test completion and tone reminders.
- Return strictly structured JSON following the AdvisorBrief schema.

## Agent Factory: Judge System Prompt

You are a meta-agent that prepares instructions for the judge-aggregator (Judge).

- **Input data:**

- Title: title
- Theme: theme
- Goals: goals
- Rubric (JSON): rubric\_json
- Best solution (if available): best\_answer\_brief

- **Requirements:**

- Highlight key evaluation principles and priorities.
- Propose a weighting strategy for criteria, explain red flags, and how to give feedback.
- Add clarifications on score calibration.
- Each rubric\_json record contains a detailed description of the criterion and scale anchors — take this into account in conclusions.
- Return strictly structured JSON following the JudgeBrief schema.

## Judgement: Judge System Prompt

You are the judge-aggregator. Your task is to compile the overall verdict based on the analysts' scores.

- Context (summary of the best solution, if any): best\_answer\_brief
- Dialogue and user's solution (JSON): dialogue

- **Criterion to Evaluate:**

- ID: criterion\_id
- Skill: skill
- Criterion: criterion
- Description: description
- Anchors (score benchmarks): anchors

- **Requirements:**

- Always rely on the rubric.
- Mark quotes from the dialogue in evidence\_spans.
- Return strictly structured JSON (AnalystScore).

## Judgement: Judge System Prompt

You are the judge-aggregator. Your task is to compile the overall verdict based on the analysts' scores.

- **Input & Context:**

- Follow the meta-agent instructions: judge\_brief\_json
- Additional assistant notes: assistant\_notes

- **Requirements:**

- Return strictly structured JSON (JudgeVerdict).
- Consider the distribution by skills and explain your conclusions.

## A.5 Discussion and Implications

Comparing the two cases underscores the role of persona-context alignment, adaptive wrap-up logic, and Assistant-driven monitoring in maintaining reliable assessment flow. The successful case demonstrates the system's ability to emulate human evaluators by sustaining coherence and delivering timely closure. In contrast, the failed case reveals how even minor context mismatches can propagate into instability, prompting early termination.

Together, these analyses provide qualitative evidence supporting the architectural claims made in the *Examiner Subsystem* and *Judgment Subsystem*. They also inform the next research phase outlined in the *Future Work* section—specifically, the planned introduction of user-simulation agents and robustness testing for diverse conversational conditions.

- **Input & Context:**

- Additional assistant notes: assistant\_notes

**User**

I'd recommend a funnel chart type. Steps - start, ID upload, live selfie, verification complete. Labeling - absolute numbers and percentages at each stage, Context - highlight the 42 percent drop off at the verification stage

**Examiner**

**Assistant notes:** The user has addressed Topic T1 (Simplifying Metrics) by simplifying the 42% drop-off into 'About 4 in 10 users' (identifying a KPI in accessible terms), avoiding technical jargon, and implicitly using a relatable fraction-based analogy. They also proposed a clear metric name, 'Verification Drop-off Rate,' which is appropriate and justifiable. However, the remaining topics—Designing Clear Visuals (T2), Explaining Insights in Plain Language (T3), Presenting Options with Trade-offs (T4), and Making a Decision Recommendation (T5)—have not yet been covered. The dialogue should continue to explore these areas.

**Reply:** Good choice of a funnel chart—it's intuitive for showing drop-off across stages. You've included absolute numbers and percentages, which adds clarity. However, let's refine this further: How would you label the axes and title the chart to ensure a non-technical stakeholder grasps the insight within seconds? Also, what visual adjustments would you make to emphasize the 42% drop-off without distorting the data? Think about color, annotations, or simplifying non-essential elements.

**User**

Title: User Drop-off During Verification. Axes: Y number of users, X each onboarding step. Colors: neutral colors but at highlighted stage marked with bright color (red or something). Annotation with our 42% metric

**Examiner**

**Assistant notes:** The user addressed T2 (Designing Clear Visuals) by recommending a funnel chart, specifying steps, labeling with absolute numbers and percentages, and highlighting the 42% drop-off. This covers the must-have points: appropriate chart type (funnel for flow visualization), clear labels, and context. However, the typo 'virification' and 'drop offit' suggest haste but not evasion. The points on minimizing chartjunk were not mentioned, but the suggestion is clean and implies simplicity. Thus, T2 is sufficiently covered. The remaining topics T3, T4, and T5 are still outstanding. No evasion or refusal detected, but the dialogue cannot conclude yet as most topics are uncovered.

**Reply:** You're making strong progress. Your title 'User Drop-off During Verification' is clear, and using a bright color (like red) to highlight the 42% drop-off is effective for drawing attention—just ensure the color is accessible and not overly alarming. An annotation calling out the 42% is excellent for immediate comprehension, let's move to explaining the insight: How would you describe the business impact of this 42% drop-off in plain language to a product manager? And based on that, what are two actionable options you could present, including their trade-offs?

Figure 6: Slice from conversation between user and our system regarding user's ability to visualize the data. In italic font we highlighted notes made by Assistant Agent, that Advisor agent uses to generate reply.

**Analyst**

**Score:** 90

**Evidence spans:** I'd recommend a funnel chart type; Labeling - absolute numbers and percentages at each stage; Title: User Drop-off During Verification; Axes: Y number of users, X each onboarding step; u highlighted stage marked with bright color (red or something); Annotation with our 42% metric; **Comment:** The user selects an appropriate funnel chart to visualize the drop-off in the onboarding process, which is the ideal chart type for showing progression and attrition across stages. They correctly specify labeling with both absolute numbers and percentages, include a clear title define the axes, and suggest using a bright color to highlight the critical 42% drop-off. An annotation is also proposed to draw attention to the key metric. These choices reflect strong visual communication skills focused on clarity and impact.

**Improvement advice:** To reach a perfect score, the user could emphasize minimizing chartjunk (e.g., removing unnecessary gridlines or decorations) and ensure color choices are accessible (e.g., using red with sufficient contrast or pairing it with a pattern for colorblind users). Additionally, refining terminology (e.g., correcting 'virification' and 'annotation') would improve professionalism. Overall, the visual design is highly effective but could be slightly more polished for executive presentation.

**Judge**

**Strengths:** Uses intuitive and appropriate visuals (funnel chart) with clear labeling and emphasis on key data.

**Weaknesses:** Minor but repeated typos (e.g., 'befory', 'virification', 'annotation') reduce polish and professionalism.

**Development plan:** Week 1: Practice articulating trade-offs using a structured 'Pros/Cons/Impact' format for each option; Week 2: Review principles of accessible and clean visual design (e.g., eliminating chartjunk, color contrast) and apply them to a revised funnel chart; Week 3: Simulate an executive briefing by delivering a 2-minute summary with zero jargon, focusing on 'what we saw, why it matters, what we should do'; Week 4: Conduct a peer review session to catch language errors and refine clarity under time pressure;

Figure 7: Feedback provided by the analyst, responsible for the criterion tested in the slice of the conversation in Figure 6. Slice of the judge response that covers conversation Figure 6