# AI-Assisted Experimentation Consulting for Educational Platform Adoption

ZACK LEE, Carnegie Learning, USA
APRIL MURPHY, Carnegie Learning, USA

Educational software teams that wish to run A/B tests within their applications often need more than access to experimentation tools: translating broad learning goals or product features into concrete experimental designs can require expert consultation over multiple sessions. This planning layer has emerged as a recurring topic when onboarding new applications to UpGrade, an A/B testing platform designed to support educational experimentation at scale. This paper describes a prototype of an AI-assisted experimentation consultant that guides users through a six-phase workflow, from gathering a description of the app through generating a report that describes testable hypotheses, outcome measures, implementation details, and how to set up the experiment in the UpGrade UI. The report can then be used as a shared artifact between researchers, developers, product managers, and other stakeholders.

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

Additional Key Words and Phrases: AI-assisted experimentation, educational A/B testing, hypothesis refinement, human-in-the-loop AI, experiment planning, UpGrade

## 1 Introduction

Adopting an educational A/B testing platform is rarely just about tool access. EdTech teams that want to run experiments in their own learning applications often need substantial expert consultation before they ever open the platform's interface. This can take the form of help deciding *what* to experiment on, *where* condition assignment should occur, *how* variants should be implemented, and *which* outcomes would make the result interpretable and actionable. This pattern has been visible in recent work supporting EdTech teams as they integrate with and onboard to UpGrade, an open-source A/B testing platform developed by Carnegie Learning [4] that is designed to address the specific constraints of embedding experiments in digital learning platforms used in real classrooms. As adoption expands to additional teams, recurring questions have arisen around topics such as where in the app the experiment should occur, condition assignment and consistency, outcome metrics, and client-side API instrumentation. While UpGrade's UI makes setting up, deploying, and managing experiments or feature flags simple, the challenge is often turning an idea–whether it involves a product feature or a traditional experimental design that must be adapted to a digital learning environment–into a concrete plan to implement and deploy in UpGrade. This broader motivation extends beyond UpGrade: educational platforms have embedded randomized experiments, including *in vivo* educational experiments, into everyday learning environments, suggesting a need for tools that help teams translate ideas into implementable experiment plans [6].

This paper presents a prototype of an **AI-assisted experimentation consultant**, the *AI Experiment Consultant*, designed to address this planning stage. The system is a web-based, chat-driven tool that guides users through a six-phase consulting workflow and produces a downloadable markdown report covering foundational concepts related to the client application, a refined hypothesis, a proposed UpGrade experiment design, a simulation summary, and implementation guidance. We illustrate this six-phase workflow with a fictional *ExampleMathApp* scenario in Section 5. This work is informed by AI-assisted hypothesis generation systems such as the AI co-scientist [1] and ExperiGen [3], but targets a narrower, human-controlled onboarding problem grounded in educational experimentation practice.

Authors' Contact Information: Zack Lee, zlee@carnegielearning.com, Carnegie Learning, Research Team, Pittsburgh, Pennsylvania, USA; April Murphy, amurphy@carnegielearning.com, Carnegie Learning, Research Team, Pittsburgh, Pennsylvania, USA.

## 2 Design Goals and Scope

The following design goals and cross-cutting principles — distinct from the six-phase consulting workflow described later — shaped the prototype, drawn from observed onboarding patterns:

(1) Begin with information gathering to establish a shared foundation for the consultation.
(2) Support ideation by suggesting candidate interventions and plausible outcome metrics.
(3) Help users refine existing ideas into specific, testable experiments.
(4) Translate the finalized hypothesis into UpGrade's vocabulary — decision point, conditions, participants, metrics.
(5) Keep humans in control through explicit approval at each major transition.
(6) Produce a structured report useful to researchers, developers, product managers, and other stakeholders.

The initial prototype focuses on the planning layer: it does not modify client code, create PRs, deploy changes, run real experiments, or make causal claims from synthetic data. It supports simple randomized designs with individual-level assignment, one decision point, and basic metrics. Future iterations of this tool could support more complex designs and explore end-to-end automation.

## 3 Prototype Overview

The prototype is a web application deployed alongside the existing UpGrade demo app [5]. The prototype source is available in a public project repository.[1] Its footprint is minimal: a login page as a soft access guard, and a chat-style main page that supports file uploads (e.g., screenshots or documents) and opens the final report in a right-side panel from which it can be copied or downloaded. The backend is built around a single primary AI request endpoint; the initial prototype does not rely on a persistent database.

UpGrade-specific knowledge is supplied through curated prompt context and report-section templates, keeping guidance consistent with what the platform actually supports. For the preflight phase, the prototype communicates with the hosted UpGrade demo backend, creating a temporary experiment, simulating participant enrollment and metric logging, and then deleting this temp data. The user is shown a conversational interface while the system maintains a parallel structured state behind the dialogue, so each phase builds on a record of the user's confirmed inputs that can be realistically matched to UpGrade's capabilities.

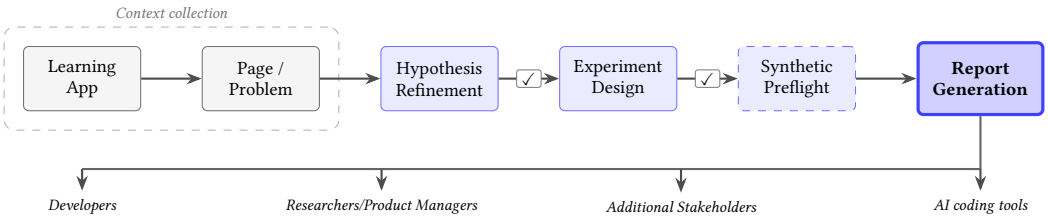


Fig. 1. Six-phase consulting workflow and report handoff. Context phases (dashed group, left) feed the AI-driven core; explicit user approvals (✓) gate the major transitions, and the dashed Synthetic Preflight node is optional. The resulting report is the shared handoff artifact across researchers, developers, stakeholders, and/or AI coding tools.

[1] Project repository: https://github.com/CarnegieLearningWeb/ai-experiment-consultant

## 4 Consulting Workflow

Figure 1 summarizes the six-phase flow, which we describe below in five grouped subsections. Context collection comes first; the later phases — hypothesis refinement, experiment design, synthetic preflight, and report generation — show how each step builds from initial information-gathering toward an implementation-ready experiment plan for UpGrade. We describe each phase in detail below.

### 4.1 Context Collection

The conversation opens by collecting basic context about the EdTech application and how it works: what the application does, who its learners are, the domain, and a description of the typical user flow. It then turns to the specific page, problem, or interaction where an experiment might run, including whether the user is trying to solve a problem, develop an intervention, or create a feature. Users can upload files such as screenshots or documents to help the AI provide more specific follow-up questions and suggestions.

### 4.2 Ideation and Hypothesis Refinement

This is the most dynamic phase. The user may arrive with a clear idea ("test whether adding a hint improves correctness rates"), an unfocused problem to solve ("students get stuck here"), or need more directed guidance on idea generation. The AI helps to refine ideas when needed, then proposes candidate interventions, providing comparison between alternative approaches if desired. During this phase, the system suggests outcome metrics based on contextual information gathered in the previous phase, then refines the approved approaches into a testable hypothesis, a sort of lightweight version of the *iterative hypothesis generation, evaluation, and refinement* pattern in the AI co-scientist model [1]. After a hypothesis is approved, the consultant offers an optional related-paper search; when the user opts in, it summarizes up to three relevant papers and possible design implications, and the selected grounding can be included in the final report.

### 4.3 UpGrade Experiment Design

Once a hypothesis is approved, the AI assembles a concrete experiment design ready to run in UpGrade with all relevant parameters: name, description, app context, decision point, conditions with their assignment weights, and metrics. The user can ask questions, request alternatives, or edit any field before approving; only the participants setting is fixed (to *Include All*) in the MVP.

### 4.4 Synthetic Preflight Simulation

With an approved design, the AI offers an optional synthetic preflight simulation against the hosted UpGrade demo backend, summarizing enrollment per condition, condition-level metric values, and basic interpretation cues. We make the caveat explicit: this is a *preflight demonstration of UpGrade mechanics*, not evidence of the intervention's effect on learning outcomes. Rather, its purpose is to help users build familiarity with UpGrade concepts and output patterns, such as enrollment by condition and metric summaries, before running a real experiment; more experienced users may choose to skip this optional step.

### 4.5 Report Generation

In the final phase, the AI assembles a detailed markdown report covering the learning app description and other relevant context, a refined hypothesis, related research grounding (if used), proposed UpGrade experiment design, simulation

```
Name:        Hint Button Experiment
Description: Tests whether adding an optional hint button
             improves correctness at first attempt on a target math problem.
App Context: example-math-app
Decision Point:
  Site:   problem_page
  Target: problem_123_hint_support
Conditions:
  control:     50%
  hint_button: 50%
Participants: Include All
Metrics:
  correctnessRate: Percent = CORRECT
  timeOnTask:     Mean
```

Fig. 2. Proposed UpGrade experiment design generated by the AI consultant for the ExampleMathApp hint-button scenario and included in the final report.

summary (if used), a recommended implementation order, and templated setup and client-integration guidance with code examples. The report represents the primary handoff artifact across user roles and AI tools (if used).

## 5 Representative Use Case and Demo

To make the workflow concrete, we walk through a fictional but representative scenario.

**Scenario.** *ExampleMathApp* is a math learning application for middle-school students. On one of the problem pages for a particular topic, the product team has noticed that many students get this problem incorrect on the first try. They want to test whether adding an optional hint button would improve correctness rates on the first attempt.

**Walkthrough.** After describing the app (including a screenshot of the problem page), the primary concern, and the targeted student population, the user works with the AI to develop and refine a hypothesis: *"Adding an optional hint button on this problem page will increase the correctness rate at first attempt without substantially increasing time-on-task."* The AI proposes correctness rate and time-on-task as metrics. The consultant also offers the optional related-paper search described above; in this walkthrough, we skip that branch and proceed directly to the UpGrade design. Once the user approves, it produces the experiment design in Figure 2.

A synthetic simulation with 200 participants returns balanced enrollment (control 97, hint_button 103), correctness-Rate 78.4 for control vs. 84.5 for hint_button, and timeOnTask 11.2 vs. 12.6. The simulation suggests that *hint_button* trends higher on correctness in this synthetic run, but notes that this is not meant to replace a true randomized experiment. The final report bundles the app and page descriptions, hypothesis, UpGrade experiment design, simulation summary with caveat, recommended implementation order, setup and experiment-creation guidance, and client-integration code examples.

The generated markdown report is the prototype's central artifact, making the experiment plan shareable across researchers, developers, and other stakeholders. As input to AI coding tools, the report can be used within the client application to support adding UpGrade API calls, implementing intervention variations in the client app, or other actions.

## 6 Discussion, Limitations, and Future Vision

The prototype described here occupies an intentionally narrow space in comparison to recent AI-assisted research tools. The AI co-scientist [1] uses tournament-style generation, debate, and evolution to autonomously traverse large, open-ended hypothesis spaces in domains like drug repurposing, while the AI Scientist [2] goes further toward end-to-end automation, from ideation through manuscript generation. While our tool borrows the high-level *iterative hypothesis generation, evaluation, and refinement* pattern from these approaches, we apply it to the tightly scoped challenge of translating an EdTech team's product or research idea into a concrete UpGrade experiment design. Where other systems maximize autonomous exploration, this AI consultant maximizes human control, ensuring that every major phase transition relies on explicit user approval, an approach that is more appropriate for users who require strong control over interventions and interpretability of outcomes.

**Limitations.** The MVP currently supports only simple randomized between-subjects designs with individual-level assignment and a single decision point. This is a deliberate scoping choice for the prototype rather than a constraint of UpGrade itself, and relaxing the single-decision-point restriction to support richer designs is the first limitation we plan to address. The synthetic preflight demonstrates UpGrade mechanics rather than learning outcomes and should not substitute for a properly powered randomized trial. Finally, the prototype has not yet been evaluated with real EdTech teams; the walkthrough in Section 5 is illustrative of a commonly encountered scenario. We anticipate being able to evaluate this tool during upcoming platform onboarding in Fall 2026.

**Future vision.** Looking ahead, in addition to supporting more complex experimental designs, the report could evolve into the hub of a human-supervised pipeline [2, 7]: a researcher approves the report; an AI coding agent drafts the variant's pull request for a developer to review and merge; automated tooling provisions the UpGrade experiment for the researcher to verify and launch; and AI-assisted analysis informs the next planning cycle. Each transition preserves human review, keeping the artifact a semi-executable specification rather than an autonomous workflow. In the near term, we will demonstrate the prototype at the workshop and pilot it with external EdTech teams during UpGrade onboarding.

## References

[1] Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, Artiom Myaskovsky, Felix Weissenberger, Keran Rong, Ryutaro Tanno, Khaled Saab, Dan Popovici, Jacob Blum, Fan Zhang, Katherine Chou, Avinatan Hassidim, Burak Gokturk, Amin Vahdat, Pushmeet Kohli, Yossi Matias, Andrew Carroll, Kavita Kulkarni, Nenad Tomasev, Yuan Guan, Vikram Dhillon, Eeshit Dhaval Vaishnav, Byron Lee, Tiago R. D. Costa, José R. Penadés, Gary Peltz, Yunhan Xu, Annalisa Pawlosky, Alan Karthikesalingam, and Vivek Natarajan. 2025. Towards an AI co-scientist. *arXiv:2502.18864.* arXiv:2502.18864 [cs.AI] doi:10.48550/arXiv.2502.18864

[2] Chris Lu, Cong Lu, Robert Tjarko Lange, Yutaro Yamada, Shengran Hu, Jakob Foerster, David Ha, and Jeff Clune. 2026. Towards end-to-end automation of AI research. *Nature* 651 (2026), 914–919. doi:10.1038/s41586-026-10265-5

[3] Jishu Sen Gupta, Syed Mohamad Tawseeq, Harini S I, Somesh Kumar Singh, Yaman Kumar Singla, David Doermann, Rajiv Ratn Shah, and Balaji Krishnamurthy. 2025. ExperiGen: Data Driven Hypothesis Generation with Experimental Verification. *OpenReview.* https://openreview.net/forum?id=LKGflZgEKB Modified 12 February 2026.

[4] UpGrade [n. d.]. UpGrade: An open-source A/B testing platform for education software. *Project website.* https://www.upgradeplatform.org/ Accessed: 2026-05-04.

[5] UpGrade Demo [n. d.]. UpGrade Demo App. *Hosted demo deployment.* https://upgrade-demo.carnegielearning.com/ Accessed: 2026-05-04.

[6] Joseph Jay Williams, Korinn Ostrow, Xiaolu Xiong, Elena L. Glassman, Juho Kim, Samuel G. Maldonado, Na Li, Justin Reich, and Neil T. Heffernan. 2015. Using and Designing Platforms for In Vivo Educational Experiments. In *Proceedings of the Second ACM Conference on Learning @ Scale (L@S 2015).* 409–412. doi:10.1145/2724660.2728704

[7] Weixian Xu, Tiantian Mi, Yixiu Liu, Yang Nan, Zhimeng Zhou, Lyumanshan Ye, Lin Zhang, Yu Qiao, and Pengfei Liu. 2026. ASI-Evolve: AI Accelerates AI. *arXiv:2603.29640.* arXiv:2603.29640 [cs.AI] https://arxiv.org/abs/2603.29640