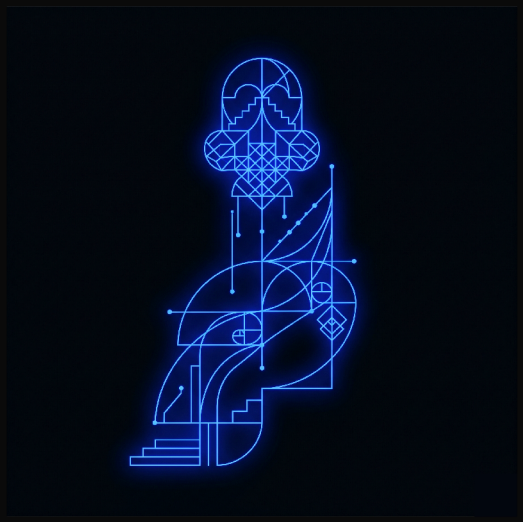
02 · TRAINING CHAPTER 2.1

From the minute of algebra to the twenty-second click

DESMOS SAT TRAINING PLATFORM

Turning the Digital SAT's graphing calculator into a trainable skill — the jump from 650 to 800 wasn't about math, it was about fluency with the tool.



STATUS	STACK	DATA / AI	QUALITY
Production · access for 14	React · Vite · Firebase	Firestore · AWS Bedrock	9 build gates

01 THE PROBLEM

A FLUENCY GAP, NOT A MASTERY GAP

The cohort analytics showed a sharp pattern: students hovering around 650 points in the Math section of the Digital SAT were not failing for lack of content. They mastered Heart of Algebra and Passport to Advanced Math — close to 70% of the exam — in their regular courses. What cost them the jump toward 800 was operational: they didn't know how to use the Desmos graphing calculator that the exam itself embeds.

A problem an expert student solves in twenty seconds of clicking took them four minutes of manual algebra, with the risk of error that drags along under time pressure.

The math curriculum they used taught how to solve, not how to use the tool the exam makes available.

It was a structural fluency gap, not one of mathematical mastery.

And no one was training it systematically — the concept was already there; what was missing was automating the procedure with the tool until it freed up working memory.

02 THE OBJECTIVE

INSTALL A REFLEX, NOT TEACH MATH

PRIMARY OBJECTIVE

Close the fluency gap: teach the student to recognize, faced with a SAT-type problem, which Desmos move solves it fastest, and to execute it without syntax hesitation.

The aim was not to teach new math, but to install a reflex: faced with a linear equation, graph it and read the root; faced with a system, find the intersection; faced with a scatter of data, write a one-line regression.

The product goal was defined in advance: that a student who enters the platform with the math already known leaves able to operate the real exam's calculator with **expert speed**.

03 THE USERS

STUDENTS IN THE 650–800 BAND

The primary users are high school students from **a K-12 school network** of AI-accelerated personalized learning, preparing for the Digital SAT and sitting in the 650 to 800 band. The production version gave access to **14 real students**; no learning outcome was measured. To anonymize tracking, each one is identified by a code — for example R-0388 — never by name.

The secondary users are instructors and administrators: through a dashboard they observe each student's progress, completion rates and the error reports that students themselves submit from each question. Role-based access control — student, admin, superadmin — keeps everyone's data isolated.

04 THE FOUNDATION

CONCEPTUAL KNOWLEDGE VS. PROCEDURAL FLUENCY

The design starts from a learning-science distinction: the difference between conceptual knowledge and procedural fluency. The high-band student already has the concept; what they lack is automating the procedure with the tool until working memory is freed.

That is why the entry unit is not mathematical but motor — **Unit 0, keystroke fluency** — to build muscle memory over fractions, exponents, subscripts and the regression tilde before loading any content. On that base, scaffolding fade was applied, the gradual withdrawal of supports as performance rises: each keystroke pattern goes through three levels — scaffold with the literal syntax in view, collapsed reminder, and blank exam like the real Bluebook environment of the SAT — and the level advances on a streak of consecutive correct answers, not on an arbitrary timer.

> *The progression is organized by cohorts according to prior score, so that each student enters at the difficulty rung that fits them and does not repeat what they already master.*

05 THE ARTIFACT

THE SPLIT VIEW: PROBLEM AND DESMOS, CO-VISIBLE

The central screen is a **split view** (split-screen): the problem on the left, the embedded and persistent Desmos calculator on the right. The decision that defines the product was recorded as an ADR in the component's own code: an app whose purpose is to teach how to use Desmos must *show* Desmos at all times. The student reads the prompt, sees the suggested move and executes it in the calculator without switching context.

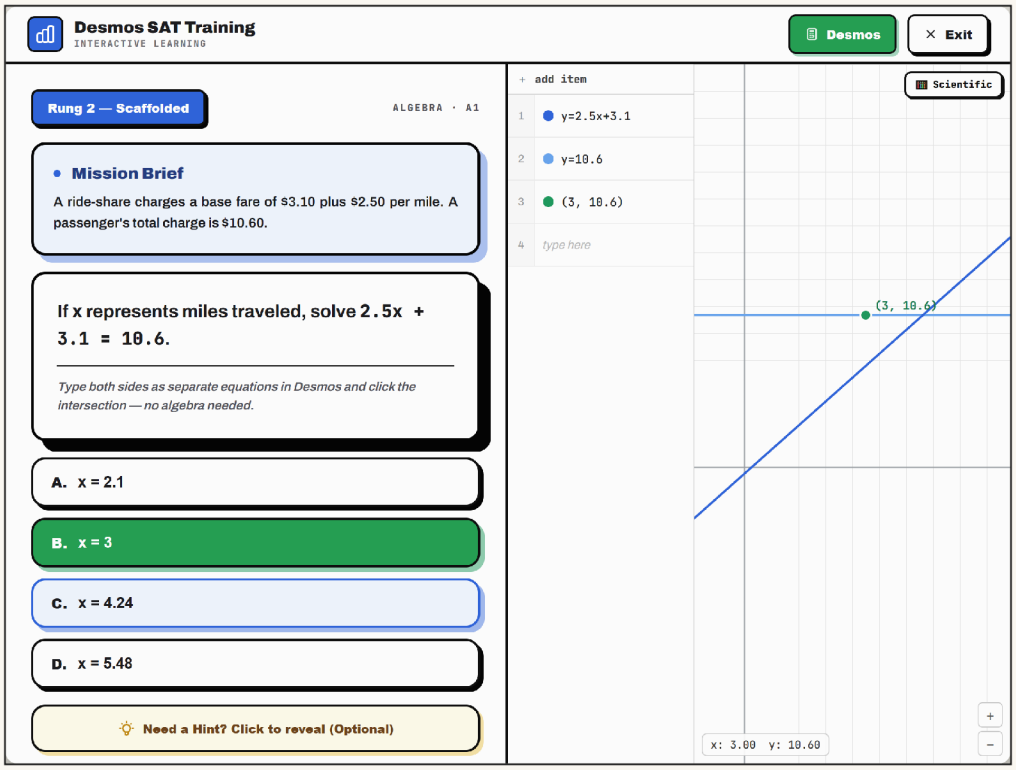


FIG. 1 — SPLIT TRAINING VIEW · SAT-TYPE PROMPT ON THE LEFT, EMBEDDED DESMOS ON THE RIGHT · THE TOOL LIVES CO-VISIBLE WITH THE PROBLEM, NEVER HIDDEN

06 THE CURRICULUM STRUCTURE

FROM UNIT 0 TO THE SAT MIX, RUNG BY RUNG

The content is structured in five layers: **Unit 0** of keystrokes; **Units 1 to 4** aligned with the exam's real domains and weights — Algebra ~35%, Advanced Math ~35%, Data Analysis ~15%, Geometry & Trig ~15%; a final timed **BOSS** that mixes the four domains with medal tiers (Platinum to Bronze, passing 4/6); and a repeatable **SAT Mix** for continuous practice.

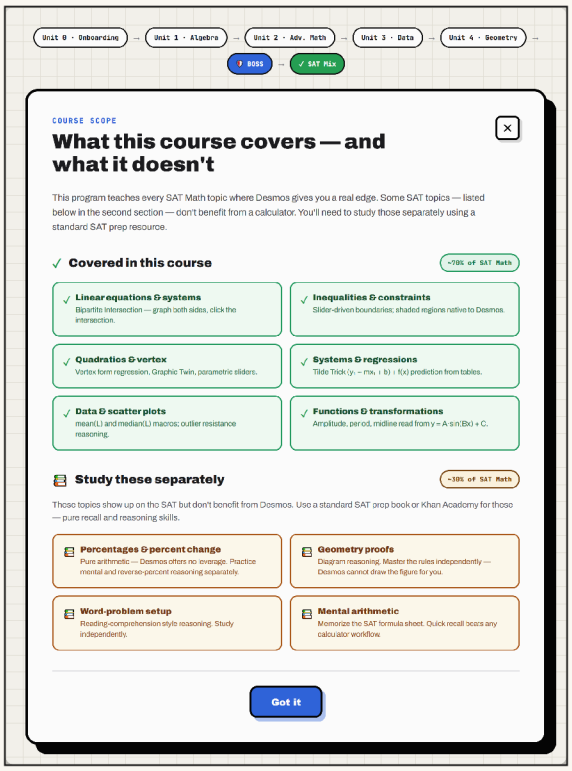


FIG. 2 — CURRICULUM MAP · THE PATH UNIT 0 → UNITS 1-4 → BOSS → SAT MIX, WITH DIFFICULTY SUB-RUNGS PER COHORT

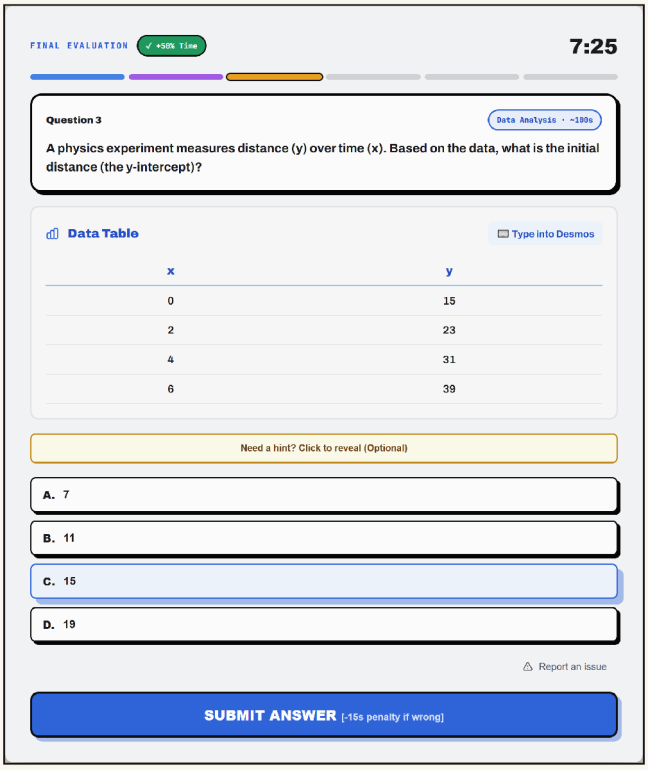


FIG. 3 — SAT MIX MOCK · REPEATABLE PRACTICE THAT REPRODUCES THE REAL BLUEBOOK ENVIRONMENT OF THE SAT — THE "BLANK EXAM" LEVEL OF THE FADE

07 THE DESIGN

CURRICULAR PROGRESSION AND INTEGRATION PIPELINE WITH THE LMS

Global state is handled with persistent Zustand and progress is synced to Firestore with a per-unit progress schema and per-student metrics. The question corpus was generated and analyzed leaning on a pipeline of research scripts that process reference SAT PDFs with models via AWS Bedrock, and is served from per-unit item banks typed in TypeScript. The platform integrates with **the network's LMS** through activity events and gradebook writes.

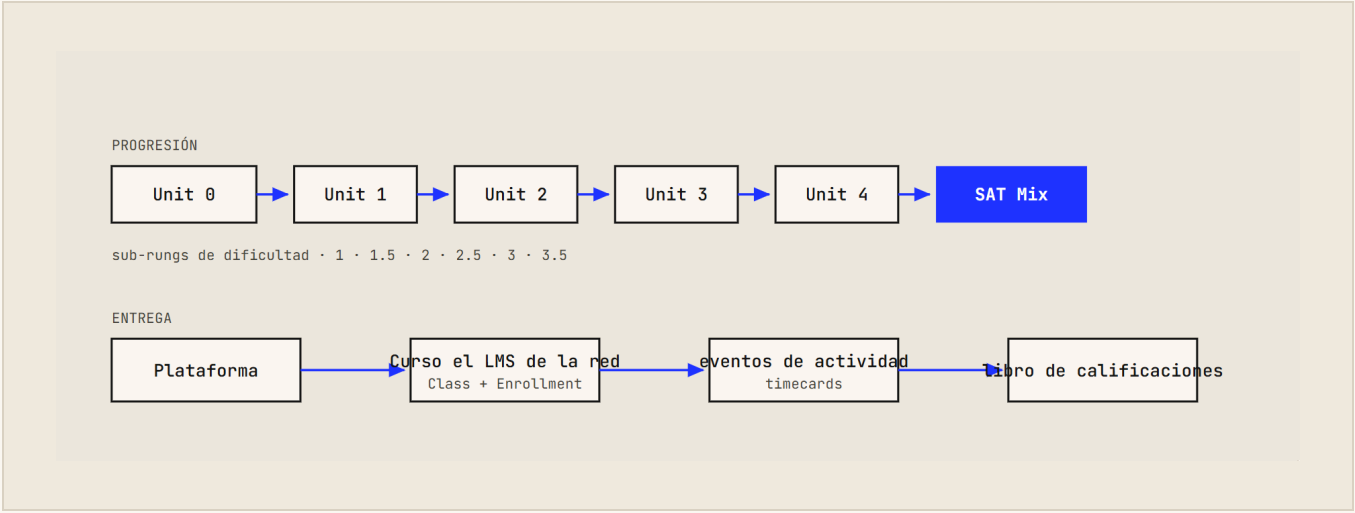


FIG. 4 — CURRICULAR PROGRESSION (UNIT 0 → SAT MIX) AND INTEGRATION PIPELINE WITH THE LMS VIA ACTIVITY EVENTS + GRADEBOOK

Desmos lives embedded and persistent — not hidden behind a floating button

Context. The initial version hid the calculator behind a floating button that covered the content. An app that teaches how to use a tool cannot hide it.

Decision. After an instructional audit it was redesigned so that Desmos lives embedded and persistent in the right panel, co-visible with the problem, inheriting the pattern of Unit 0.

Accepted trade-off. Less horizontal space for the prompt, but the tool stays always in view — and showing the tool *is* the product's pedagogy.

08 CONSTRUCTION AND VALIDATION

QUALITY INSTITUTIONALIZED AS BUILD GATES

Development ran with an AI-first flow sustained over about three and a half weeks. Content quality does not rest on manual review: it was institutionalized as a set of automatic gates that run before each build — a fidelity check that recomputes the answer from the item's data, strict skill coverage, option ordering, LaTeX render, valid Desmos syntax, keystroke patterns, and correspondence between question type and move. If any one fails, the build fails.

Validation combined three fronts: **mathematical peer review** with two high school experts, who verified the correctness of the items and the pedagogy of the moves; **instructional QC** that audited the learning experience end to end; and the battery of **automatic gates** as a permanent safety net. Deployment is to Vercel with auto-deploy from the main branch and an invitation system with superadmin approval. The students themselves feed an error-report channel from each question.

09 THE FOUNDATION, TURNED INTO A TABLE

THE PRODUCT'S EVIDENCE

Students with access	14 · no outcome measurement
Construction window (v2)	May 25 → Jun 18 2026
Commits (v2)	346
Content units	Unit 0 (keystrokes) + Units 1-4 + BOSS + SAT Mix
Cohorts by prior score	foundational <600 · intermediate 600-740 · advanced 750-800
Quality gates at build	9 (fidelity, coverage, ordering, render, desmos, keystrokes, repetition, move, move coverage)
Scaffolding fade levels	3 (scaffold · reminder · exam)
Weight of SAT domains covered	Algebra ~35% · Adv Math ~35% · Data ~15% · Geo&Trig ~15%
Institutional integration	the network's LMS — activity events + gradebook writes
Validation	mathematical peer review (2 experts) + instructional QC
Status	In production

10 THE RESULT

FROM AN ANALYTICS INTUITION TO AN OPERATIONAL PRODUCT

The platform went to production with open access for 14 students, covering the full path from Unit 0 to BOSS plus the SAT Mix, with LMS integration active. Validation was of judgment, not of outcome: measuring the sustained effect on the

real score would have required a rollout and a window I did not control. What was measured was product quality — and the peer review was unequivocal: one reviewer described having learned a lot about Desmos with the system itself, and another rated it a very solid course.

Beyond the deployment, the underlying result was turning an analytics intuition Ch. 1 — the 650 to 800 gap is one of Desmos, not of math — into an operational product with a curriculum of verified moves, a scaffolding-fade engine and a set of quality gates that ensure no incorrect item reaches the student.

11 LESSONS

WHAT THE PROJECT LEFT BEHIND

- **Design.** An app that teaches a tool has to display that tool, not hide it; the audit that moved Desmos from a floating overlay that covered content to a persistent embedded panel changed the experience at its root.
- **Trust in the content.** Machine-verified does not equal fully verified — after regenerating content it pays to look at it rendered, because the eye catches legibility problems and clones the gate does not see.
- **Architectural.** Encapsulating the data layer well pays off: having Firestore isolated behind a service layer made it possible to later evaluate and execute a backend migration with a clean cutover.
- **Process.** Institutionalizing quality as gates that fail the build, instead of human checklists, is what sustains a pace of hundreds of commits without the content degrading.