

AI Tutor Adaptive Quiz Engine: Technical & Architectural Specification

Espranza Innovations

July 2026

Abstract

This document outlines the technical architecture of the AI Tutor Adaptive Quiz Engine. It details the process of generating question banks from chapter textbooks, managing the student's conversational learning phases, selecting appropriate questions dynamically to match student skill levels, updating student mastery based on performance and speed, and directing struggling students to prerequisite review detours when conceptual gaps are identified.

Contents

1	Introduction & System Overview	3
2	Stage 1: Question Pool Generation Pipeline	3
2.1	Semantic Retrieval (RAG)	3
2.2	Cognitive Taxonomy & Difficulty Levels	3
2.3	Subject-Specific Calibrations	3
2.4	Distractors and Misconception Tagging	4
3	Stage 2: Student Conversational Chat Flow	4
3.1	Conversational Teaching Phases	4
3.2	Content-Based Pace Gating	4
3.3	Engagement Tracking and Toning	5
3.4	Stall Prevention (Boredom Escape)	5
3.5	Confusion Gating	5
3.6	Whiteboard Diagram (Asset) Integration	5
4	Stage 3: Adaptive Quiz Selection Engine	5
4.1	Attempt Inception	5
4.2	The Selection Strategy	6
4.2.1	Difficulty Window Matching	6
4.2.2	Cascading Fallbacks	6
4.3	Subjective Quota Gating (Mixed Mode)	6
5	Stage 4: Evaluation, Mastery Updates & Gating	6
5.1	Grader Evaluation	6
5.2	Mastery Score Updates	6
5.3	Topic Gating Criteria	7
5.4	Dashboard Progress Metrics	7

6	Stage 5: Prerequisite Remedial Detour Engine	7
6.1	Struggle Detection	7
6.2	Prerequisite Resolution & Detour Routing	8

1 Introduction & System Overview

The AI Tutor platform utilizes a personalized, self-calibrating student assessment framework. Rather than presenting fixed quizzes, the platform automatically generates a comprehensive pool of questions grounded directly in textbook content.

During an active session, an adaptive engine matches question difficulty with the student’s estimated ability in real-time. This dynamic customization optimizes the student’s learning efficiency, ensuring questions are challenging enough to drive growth but not so difficult as to cause frustration.

2 Stage 1: Question Pool Generation Pipeline

The master question bank for each chapter is generated automatically upon chapter ingestion by the `ChapterQuizService`. The pipeline uses semantic retrieval (RAG) and calls Claude to synthesize balanced assessment items.

2.1 Semantic Retrieval (RAG)

To generate questions for a specific topic, the service performs a semantic similarity search. It compares the topic title with all textbook content chunks saved in the database. The top 5 most similar chunks (that fall within a strict distance threshold) are retrieved.

These textbook passages are supplied to the AI generator to serve as the ground truth. This ensures that every question is factually accurate, relevant, and strictly bounded by the uploaded curriculum (eliminating hallucinations).

2.2 Cognitive Taxonomy & Difficulty Levels

The AI generator is instructed to generate exactly **25 questions per topic**, distributed evenly across five cognitive difficulty bands aligned with Bloom’s Taxonomy:

Cognitive Level	Cognitive Type	Difficulty Score	Expected Response Time
Level 1	Recall	1.0 – 1.9	8–15 seconds
Level 2	Understanding	2.0 – 2.9	15–25 seconds
Level 3	Application	3.0 – 3.9	25–40 seconds
Level 4	Reasoning	4.0 – 4.4	40–60 seconds
Level 5	Analysis	4.5 – 5.0	60–90 seconds

Table 1: Cognitive Level and Difficulty Score Mappings

2.3 Subject-Specific Calibrations

The pipeline customizes prompt instructions based on the subject name:

- **Mathematics:** Mandates balanced equations written in LaTeX format. Distractors must target logical operational mistakes, such as distribution errors, sign flips, or incorrect order of operations (PEMDAS).
- **Science:** Requires SI units of measurement. Distractors must target common scientific misconceptions (such as confusing mass with weight, or photosynthesis with respiration).
- **Language & Humanities:** Focuses on vocabulary, textual comprehension, and primary-source synthesis.

2.4 Distractors and Misconception Tagging

Questions are structured with diagnostic tags:

- **Multiple Choice Questions (MCQ):** Built with 4 options. Every wrong option (distractor) contains a specific misconception tag describing the conceptual error (e.g., "*Confused seasons with daily cycles*").
- **Short Answer Questions:** Free-text questions with a detailed model answer and a list of expected misconception tags.

3 Stage 2: Student Conversational Chat Flow

The student learning session is driven by a cognitive teaching state machine. The tutor does not immediately quiz the student; instead, it progresses through conversational teaching phases to ensure deep comprehension before assessment.

3.1 Conversational Teaching Phases

For each topic, the tutor guides the student through the following chronological phases:

1. **INTRO (Introduction):** Introduces the core concept briefly in a friendly tone (usually 2–3 sentences) and prompts the student with a simple opening question.
2. **DEEPENING:** Dives into the underlying details, mechanisms, and rules of the concept to expand the student’s understanding.
3. **REAL_WORLD:** Connects the concept to a relatable real-world analogy or practical application to solidify retention.
4. **QUIZ_PENDING:** Triggers the topic assessment by displaying one quiz question (MCQ or short-answer) on the screen. During this phase, follow-up questions are disabled so the student remains focused on the assessment.
5. **QUIZ_RECOVERY:** If the student answers the quiz incorrectly, the tutor enters this recovery phase. It explains the mistake, re-teaches the concept using new examples, and assesses when the student is ready to try again.
6. **RECOVERY QUIZ_PENDING:** Once the student confirms readiness, the tutor administers a new re-test question to verify recovery.

3.2 Content-Based Pace Gating

To prevent rushing through topics, the state machine requires a minimum number of high-quality conversation turns in the teaching phases before allowing advancement. This pace is governed by the length of the textbook topic content:

- **Simple Content** (under 300 characters): Requires at least 1 turn each in the INTRO, DEEPENING, and REAL_WORLD phases.
- **Medium Content** (300–700 characters): Requires at least 1 turn in INTRO, 2 turns in DEEPENING, and 1 turn in REAL_WORLD.
- **Complex Content** (over 700 characters): Requires at least 1 turn in INTRO, 3 turns in DEEPENING, and 2 turns in REAL_WORLD.

3.3 Engagement Tracking and Toning

The system scores the quality of student messages on a rolling scale of 0 to 100. Each turn, the AI categorizes the student’s response to adjust this score:

- **HIGH** (+20 points): Student asks a curious question or explains the concept in their own words.
- **MEDIUM** (+10 points): Student gives a normal on-topic response (e.g., 4+ words) or asks for clarification.
- **LOW** (−10 points): Student gives a lazy, short reply (e.g., "ok", "yes", or single emoji).
- **SPAM** (−10 points): Student sends completely off-topic or gibberish messages.

Based on this rolling score, the tutor adapts its response length and tone:

- **Emergency Mode** (score under 30): The student is disengaged. The tutor keeps its responses strictly to 1–2 sentences and asks direct, simple choice questions (like "A or B?") to draw them back.
- **Low Engagement** (score under 60): The tutor keeps answers short and punchy, relying on relatable analogies to spark interest.
- **High Engagement** (score 60 or higher): The student is active. The tutor goes deeper into details with enthusiasm.

3.4 Stall Prevention (Boredom Escape)

If a student provides 5 consecutive low-quality replies (LOW or SPAM), the state machine triggers an escape hatch and force-advances to the next phase. This prevents disengaged students from getting stuck in a repetitive loop.

3.5 Confusion Gating

If the student explicitly states they do not understand, asks for a re-explanation, or gives a conceptually incorrect answer, the AI sets a confusion flag. This flag temporarily blocks the student from advancing to the next teaching phase, keeping them in the current phase until the tutor resolves their doubt.

3.6 Whiteboard Diagram (Asset) Integration

The system integrates visual learning by showing relevant diagrams on the student’s whiteboard. The engine enforces a rule of at most one diagram shown per teaching phase. The diagrams are pulled from the topic assets, resolved to presigned S3 URLs, and displayed alongside conceptual bullet points on the frontend.

4 Stage 3: Adaptive Quiz Selection Engine

When a student starts an adaptive quiz, the engine serves questions sequentially, targeting the student’s estimated ability level and updating it dynamically.

4.1 Attempt Inception

Upon starting a quiz attempt, the engine identifies the first topic in the chapter (using its order index) and fetches the student’s mastery score for that topic. New topics are initialized to a default starting score of **3.0** on a scale of 1.0 to 5.0.

4.2 The Selection Strategy

To select the next question, the engine employs a target difficulty window:

4.2.1 Difficulty Window Matching

The engine targets questions whose difficulty score is within **0.8 units** above or below the student's current mastery score. For example, if a student's mastery score is 3.2, the target difficulty window is 2.4 to 4.0.

To prevent the student from getting the exact same question if they refresh the page, the engine performs a **weighted random selection** from all unanswered questions in this window. Questions whose difficulty matches the student's score more closely are given a higher probability of selection.

4.2.2 Cascading Fallbacks

If the target difficulty window contains no unanswered questions, the engine cascades through fallbacks to ensure the quiz never gets stuck:

1. Pick any unanswered active question in the current topic at random.
2. If the current topic is out of unanswered questions, pick any unanswered active question in the entire chapter at random.

4.3 Subjective Quota Gating (Mixed Mode)

In mixed quizzes (MCQ + Short Answer), the engine guarantees a minimum quota of subjective questions (at least 3, or one-third of the total quiz size). The engine monitors the number of short-answer questions answered and injects them at regular intervals if the quota has not been met.

5 Stage 4: Evaluation, Mastery Updates & Gating

5.1 Grader Evaluation

When a student submits an answer:

- **MCQ:** Evaluated instantly. If incorrect, the misconception tag linked to the chosen option is recorded.
- **Short Answer:** The text response is sent to Gemini alongside the model answer and chapter context. Gemini determines correctness, generates descriptive markdown feedback, and provides a misconception label if incorrect.

5.2 Mastery Score Updates

The student's topic mastery score shifts dynamically after each answer:

- ****Correct Answer:**** Increases the student's mastery score.
- ****Incorrect Answer:**** Decreases the student's mastery score.

The baseline adjustment step is 0.5. However, this is scaled by a ****time multiplier**** based on how fast the student answered relative to the question's expected response time:

- **Very Fast** ($\leq 50\%$ of expected time): Adjusted by **1.5x** (reflects high fluency).

- **Normal** ($\leq 150\%$ of expected time): Adjusted by **1.0x**.
- **Slow** ($\leq 300\%$ of expected time): Adjusted by **0.5x** (reflects minor processing hesitation).
- **Very Slow** ($> 300\%$ of expected time): Adjusted by **0.1x** (prevents heavy penalization for slow typing or rewarding lucky guesses on long pauses).

5.3 Topic Gating Criteria

To complete the active topic and advance to the next, the student must meet subject-aware gating rules:

- **Language Subjects:** Requires at least **1** consecutive correct answer and a mastery score of **3.0** or higher.
- **Math and Science Subjects:** Requires at least **2** consecutive correct answers, at least **3** attempts, and a mastery score of **3.5** or higher.

Once these conditions are met, the topic is marked completed, and the quiz shifts to questions from the next topic. If the student hits the total quiz size limit (e.g., 10 questions) before mastering all topics, the quiz completes.

5.4 Dashboard Progress Metrics

The platform aggregates student activity into two separate dashboard metrics:

- **Learn Meter:** The simple percentage of topics in the chapter that the student has successfully completed. This metric only increases.
- **Concept Retention:** Represents the student's average mastery score across the chapter's topics, scaled from 0% to 100%. To ensure the metric accurately reflects overall chapter progress, unattempted topics are counted as 0% retention.

6 Stage 5: Prerequisite Remedial Detour Engine

Upon completing an adaptive quiz, the system checks if the student is fundamentally struggling, which triggers a background review process.

6.1 Struggle Detection

A student is flagged as confused and eligible for a remedial detour if:

- They answered **3** or more short-answer questions incorrectly in the quiz.
- Their average mastery score across all attempted topics in the chapter is **2.5** or lower.

If these conditions are met, the system increments a confusion counter. Once the counter crosses the threshold, the system initiates a background task to find an appropriate prerequisite chapter detour.

6.2 Prerequisite Resolution & Detour Routing

The background task searches for prerequisite chapters mapped to the active chapter. If no explicit connections exist, it falls back to all other chapters in the same book.

It filters out any chapters the student has already completed and ranks the remaining candidates based on subject and grade proximity (prioritizing same-subject, same-grade chapters first).

Once the best prerequisite chapter is identified, the system uses Gemini to generate an encouraging welcome message.

- If the student **accepts** the suggestion, the backend creates a new learning session targeting the prerequisite chapter and seeds the welcome message.
- If the student **declines** the suggestion, the confusion counter resets to 0 and the suggestion is cleared.

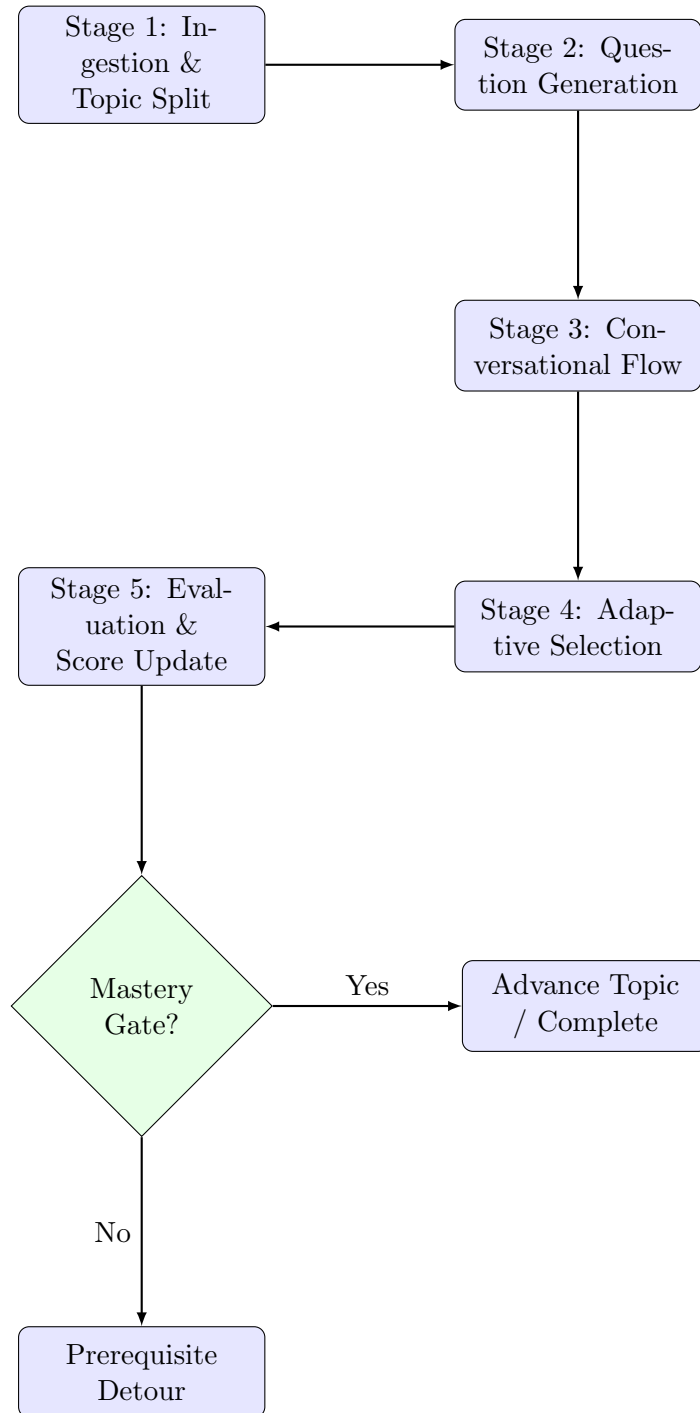


Figure 1: AI Tutor Lifecycle and Information Flow