

# Technical Assessment: AI Reading Comprehension Interface

## The Challenge

We're rebuilding our reading comprehension experience from the ground up. The current interface is functional, but students find it uninspiring and too easy to game. We want you to design and build the **next-generation reading comprehension interface** that will become our primary learning tool.

## Context

A big part of the EdAccelerator English Program is helping students improve their English comprehension skills.

Imagine the following scenario: Students currently complete reading passages with multiple-choice questions, but engagement is dropping and learning outcomes are not where we want them.

You are required to build an **updated and improved reading comprehension interface**. We are giving you user feedback, a sample passage, and creative freedom to build something better.

## User Feedback

"It's annoying seeing the entire passage all at once. Is it possible to break it apart? I know we can't change the length of the passage."

"Multiple choice is too easy, I can just guess the answer. Can you make it so we have to type?"

"I finish the questions and immediately forget what I read. There's no reason to actually understand the passage."

"When I get an answer wrong, I don't really learn why. The explanation is there but I just click continue."

"It feels like a test, not like learning. I dread doing these."

"Sometimes I want to go back and re-read a specific part but I can't find it quickly."

"My younger brother uses the app too and he's way slower at reading. The same interface doesn't work for both of us."

## Requirements

### Must Have

- Display reading passages with AI-generated questions
- Use AI to generate high-quality comprehension questions from the passage
- Track correct and incorrect answers
- Show a final score at completion
- Work on both mobile and desktop

### AI Question Generation

- You are responsible for generating the questions using AI (OpenAI, Anthropic, or similar)
- Questions should test genuine comprehension, not surface-level recall
- Consider different question types and difficulty levels
- Question quality matters and should lead to strong learning outcomes
- You may generate questions at build time, on-demand, or using a hybrid approach

### Mock Data

Use the following passage for the assessment. You must generate the questions using AI. This is the raw passage only.

```
{ "passage": { "id": "passage-1", "title": "The Secret Life of Honeybees", "content": "Inside every beehive, there is a world more organized than most human cities. A single hive can contain up to 60,000 bees, and every single one has a job to do. At the center of the hive is the queen bee. She is the only bee that lays eggs—up to 2,000 per day during summer. Despite her title, the queen doesn't actually make decisions for the hive. Her main job is simply to lay eggs and keep the colony growing. The worker bees are all female, and they do everything else. Young workers stay inside the hive, cleaning cells, feeding larvae, and building honeycomb from wax they produce from their own bodies. As they get older, they graduate to guarding the hive entrance. The oldest workers become foragers, flying up to five miles from the hive to collect nectar and pollen. Male bees are called drones. They don't collect food, don't guard the hive, and don't have stingers. Their only purpose is to mate with queens from other hives. In autumn, when food becomes scarce, the workers push the drones out of the hive to conserve resources. Bees communicate through dancing. When a forager finds a good source of flowers, she returns to the hive and performs a 'waggle dance' that tells other bees exactly where to find the food. The angle of her dance shows the direction relative to the sun, and the length of her waggle shows the distance. This tiny insect has been making honey the same way for over 100 million years. Every spoonful of honey represents the life's work of about twelve bees." } }
```

## Everything Else

Up to you. We care about your product thinking as much as your code.

## Technical Constraints

Framework	Next.js 14+ (App Router)
Language	TypeScript
Styling	Tailwind CSS
AI	OpenAI, Anthropic, or similar (use your own API key)
Deployment	Vercel (provide a live link)

## Submission

1. GitHub repository (public, or add @edaccelerator as a collaborator)
2. Live demo deployed on Vercel (please ensure it is publicly accessible and does not require requesting access)
3. README explaining interpretation of feedback, AI question generation approach, key decisions, improvements, and time spent

## Questions

Reply if anything is unclear. We are happy to clarify constraints but will not provide design direction as that is part of the evaluation.