AMERICAN UNIVERSITY OF BEIRUT

ASSESSLY: AN INTELLIGENT PLATFORM FOR READING ENGAGEMENT AND DIFFERENTIATED ASSESSMENT IN HIGHER EDUCATION

by
SILAS SANGMIN

A project
submitted in partial fulfillment of the requirements
for the degree of Master of Arts
in Computing in Education
to the Department of Education
of the Faculty of Arts and Sciences
at the American University of Beirut

Beirut, Lebanon
October 2026

AMERICAN UNIVERSITY OF BEIRUT

ASSESSLY: AN INTELLIGENT PLATFORM FOR READING ENGAGEMENT AND DIFFERENTIATED ASSESSMENT IN HIGHER EDUCATION

by

SILAS SANGMIN

Approved by

Dr. Hoda Baytiyeh, Associate Professor Department of Education	First Reader
Dr. Saeed Raheel, Lecturer Department of Computer Science	Second Reader

Date of project presentation: October 19, 2026

ACKNOWLEDGEMENTS

I thank Almighty God for giving me the strength and power to complete this degree program.

I sincerely thank my supervisors, Professor Hoda Baytiyeh and Dr. Saeed Raheel, for their support and thoughtful comments throughout the project; their advice was especially valuable when things were difficult.

I should also thank the 11 volunteers who spent their time testing Assessly as part of the 14-day field pilot; your efforts shaped this platform.

To conclude, I thank my family and friends for their continuous support and encouragement at every stage of this journey.

ABSTRACT
OF THE PROJECT OF

Silas Sangmin for Master of Arts
Major: Computing Education program

Title: Assessly: An Intelligent Platform for Reading Engagement and Differentiated Assessment in Higher Education

Reading non-compliance occurs when students do not engage properly with assigned course materials. This is an issue in higher education, especially in the current age of artificial intelligence. To check whether students have engaged with the reading material, teachers usually rely on static quizzes. But students can easily bypass these quizzes by sharing answers and using generative models. Also, instructors cannot see what actually happens during the reading process. To address this problem, the paper introduces Assessly. A web-based platform that shifts the focus from simply verifying that reading has taken place through quizzes to making the reading process visible, interactive, and meaningful. Assessly integrates with Moodle via the Learning Tools Interoperability (LTI) 1.3 standard and includes an engagement layer that wraps PDF readings uploaded by instructors. This layer provides social annotation with the ability to ask, comment, or reply to colleagues on the reading document and an anonymity feature that keeps both the students' and the instructors' identities hidden. Other features of the system include active-recall flashcards that ask for typed answers and are graded by a large language model; a Socratic tutor that is implemented to responding only with guiding questions based on the reading material; and per-attempt quizzes that are drawn from an AI-generated question bank, administered in a soft-lockdown environment with individual time allocations set for each student.

Assessly uses behavioral indicators, such as the annotations, the results from flashcards and quizzes, and the way users interact with the tutor, to provide information on the instructor dashboard at the class level and to produce personal recommendations. A combined engagement score is kept in sync with the Moodle gradebook (40% from quizzes, 30% from flashcards, and 30% from annotations). This ensures the assessment reflects what happens during the reading process rather than relying solely on recall. This method is based on self-regulated learning theory and the ICAP framework. A 14-day field trial with 11 participants was conducted from six higher-education institutions to test the system. All the participants used at least 2 of the 4 channels; 73% used 3 or more. 7 of the 11 people who completed the survey said they intended to continue using the platform in an actual course. The combined grade could distinguish between active reading and performance based only on quizzes, confirming the effectiveness of the cross-signal infrastructure. However, evaluating learning outcomes will require future controlled studies.

Keywords: Reading non-compliance, Reading engagement, social annotation, Retrieval practice, Socratic tutoring, Cross-signal personalization, ICAP, self-regulated learning, LTI 1.3, large language models.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS 1

ABSTRACT 2

ILLUSTRATIONS 8

TABLES 9

ABBREVIATIONS 10

INTRODUCTION 11

Background and Motivation 11

Problem Statement 12

Objectives 13

Scope and Limitations 14

Report Organization 15

LITERATURE REVIEW AND THEORETICAL FRAMEWORK 16

Reading Non-Compliance in Higher Education 16

Social Annotation 16

Retrieval Practice and Active Recall 17

Artificial Intelligence in Reading and Assessment 18

Automatic Question Generation 18

Automated Scoring of Open Responses 18
Dialogue Tutoring 18
Feedback and Formative Assessment 19
LMS Integration Standards 19
Theoretical Framework 20
The ICAP Framework 20
Self-Regulated Learning 21
SYSTEM DESIGN 22
Design Principles and Requirements 22
Architecture 22
Data Model 23
The Engagement Layer 24
Social Annotation 24
Active-Recall Flashcards 24
The Socratic Tutor 25
Cross-Signal Personalization 25
Composite Engagement Grading 26
AI Service Design 27
LMS Integration and Grade Passback 28
Assessment Integrity and Accessibility 29
IMPLEMENTATION 31
Backend 31

Frontend 31

Deployment 32

Privacy and Security 32

TESTING AND EVALUATION 33

Automated Testing 33

Field Pilot with Real Users 33

Method 34

Engagement and Channel Adoption 34

Composite Grades and the Cross-Signal Claim 35

Survey Results 36

Qualitative Findings 38

Defects Found and Fixed During the Pilot 39

Threats to Validity 40

Evaluation Against Objectives 41

CONCLUSION AND FUTURE WORK 42

Summary 42

Contributions 42

Limitations 43

Future Work 44

Controlled Efficacy Study 44

Tutor Improvements 45

Support for Additional Document Types and Video 45

APPENDIX 46

REFERENCES 51

ILLUSTRATIONS

Figure

1. System Architecture of the Assessly Platform 23
2. Cross-Signal Data Flow From the Transactional Tables to the My Next Step and Class Pulse Surfaces 26
3. LTI 1.3 Launch, Activity Binding, and Grade Passback Sequence 29
4. Signal-Channel Adoption During the Field Pilot (n = 11) 35
5. Composite Grade Composition Per Participant (N = 11) 36
6. Per-Feature Usefulness Ratings from the Final Survey(N=9) 38
7. Retention Intent Among Survey Respondents (n = 9) 38

TABLES

Table

1. Per-Feature Usefulness Ratings (n = 9, scale 1 to 5) 37
2. Production Defects Found and Fixed During the Pilot 39

ABBREVIATIONS

AI - Artificial Intelligence

API - Application Programming Interface

CDN - Content Delivery Network

CORS - Cross-Origin Resource Sharing

CRUD - Create, Read, Update, Delete

CSRF - Cross-Site Request Forgery

HTTP - Hypertext Transfer Protocol

ICAP - Interactive, Constructive, Active, Passive

JSON - JavaScript Object Notation

JWT - JSON Web Token

LLM - Large Language Model

LMS - Learning Management System

LTI - Learning Tools Interoperability

OIDC - OpenID Connect

PDF - Portable Document Format

REST - Representational State Transfer

SD - Standard Deviation

SQL - Structured Query Language

SUS - System Usability Scale

CHAPTER 1

INTRODUCTION

Background and Motivation

Assigned reading forms the core of most university courses, but studies frequently indicate that only a small number of students complete it. Kerr and Frese (2017) state that only 20 to 30% of undergraduate students complete the required reading, and St Clair-Thompson, Graham and Marsham (2018) found that students spend less time reading academic texts than the university requires. This situation is steady and not occasional; undergraduates consider reading to be important even though they do not carry it out, seeing it as too lengthy and not sufficiently linked to the outcomes of the classes (Gorzycki, Desa, Howard & Allen, 2020), and teachers either lecture on the material, thereby removing any incentive to read it, or they keep an eye on students' reading through the use of quizzes.

Students can avoid static reading quizzes based on fixed question banks by sharing answers, using search engines, and, more recently, using generative AI tools that can answer comprehension questions without the student opening the text. A quiz score reflects only the result of reading; it says nothing about how the student read. For example, whether they questioned the text, discussed it with their classmates, checked their own retention, or asked for help when they were confused. Two students who get the same quiz score might have had completely different experiences with the material, and the instructor will not be able to tell them apart. The learning sciences literature is just as clear in stating that the kinds of behavior that quizzes fail to capture are the ones that actually matter: engaging with the material in a generative and interactive way leads

to better learning than being passively exposed to it (Chi et al., 2018), and retrieval practice results in more lasting learning than restudying (Adesope, Trevisan & Sundararajan, 2017; Agarwal, Nunes & Blunt, 2021).

The project thus examines non-compliance from the other side. Rather than developing a more effective lie detector for determining if reading had taken place, Assessly makes the act of reading visible, social, and worthwhile: it makes it visible since each interaction leaves a behavioral signal which the platform can collect for both the student and the instructor; it makes it social because annotation turns private reading into a low-stakes discussion within the class; and it makes it worthwhile since the grade is rewritten in such a way that the engagement behaviors account for most of the marks.

Problem Statement

Current tools often support only part of the reading process. While learning management systems include gradebooks and course information, they generally deliver assigned readings as only reading readable files. Engagement tools specialized in functions such as annotation or flashcards work on their own and do not give a full picture of student engagement. Quiz tools designed for assessment can easily be avoided and offer only limited insight into true comprehension.

The gap is a platform that brings three functions together. A platform that enables (a) the act of reading to be monitored through various behavioral channels as part of the institution's LMS workflow; (b) provide channels useful to both the student and the instructor without requiring a sophisticated learner-modeling system; and (c) the incentive structure, in the form of the grade, to correspond with the kind of engagement that the learning sciences consider to be productive.

Objectives

During the design, implementation, and evaluation of the Assessly platform, the project aims to achieve the following objectives:

1. Create a reading engagement layer for instructor-provided PDFs, including social annotation with optional anonymity and threaded replies. AI grades these active-recall flashcards, annotations quizzes and requires typed responses, along with a Socratic tutor limited to asking questions that guide the student rather than answering.
2. Introduce cross-signal personalization. The transactional model data will be combined using SQL to provide class-level information for an instructor's dashboard and to make specific next-step recommendations for individual students, eliminating the need for a separate learner-modeling service. The signals are quiz results, the tutor's actions, and activity related to flashcards and annotations.
3. A grading synergy will weigh the reading process the same as the system grade, at 40/30/30 for quiz, flashcards, and annotations, respectively. This grade is calculated when the quiz is submitted, remains aligned with the learning management system gradebook, and updates as more engagement data is recorded.
4. An AI content pipeline uses a dual-provider architecture to ensure resilience; based on the provided text, it generates Socratic tutoring, flashcard decks, and question pools for each document, with prompt-level rules ensuring the output is substantial.
5. The integration of LTI 1.3 with Moodle includes single sign-on, automatic provisioning of users and courses, durable links between LMS activities and platform assignments, and reliable pass-back of grades in accordance with the Assignment and Grade Services specification.

6. Safeguards commensurate with the need to maintain the integrity of assessments, for example by selecting the questions for each attempt from a larger pool, by allocating a specific amount of time to each student for access to the assessment, and by using a soft-lockdown system for quizzes which keeps a record of any violations for audit purposes.
7. A review of the platform after it had been put into use, carried out through a field pilot involving the actual users on the production system and including some automated testing, to evaluate usability and feature adoption and to check whether these users are in fact generating the kind of cross-signal data on which the grading and personalization are based.

Scope and Limitations

This section pertains to the design, implementation, and deployment of Assessly as a reading engagement tool for higher education institutions, with Moodle serving as the LMS of choice to be integrated with, as well as the platform's formative evaluation. In the course of the field pilot, we have gauged infrastructure validity, adoption, and usability. With a comparison-group design, one could assess learning outcomes or reading compliance in a controlled setting, but that is not part of this formative evaluation. As for the quality of any content produced by AI, it is only as good as the third-party large language models allow. Structured prompts and review from the instructor go some way to curbing the risk of hallucination but do not rule it out. The lockdown environment is meant to be an audit and a deterrent, not a means of prevention, a matter of candor addressed in Chapter 3.

Report Organization

The literature is examined in Chapter 2, covering topics such as reading non-compliance, social annotation and retrieval practice, the use of AI in an educational context, and integration standards; this chapter also establishes the two theoretical frameworks. As for the system design, that is found in Chapter 3. Implementation is detailed in Chapter 4. Testing, along with the results of the field pilot, is reported in Chapter 5. The report concludes in Chapter 6 with contributions, limitations, and an outline of future work.

CHAPTER 2

LITERATURE REVIEW AND THEORETICAL FRAMEWORK

Reading Non-Compliance in Higher Education

One of the more reliably reproduced results in higher education research is that of reading non-compliance. In a 2017 survey of the literature, Kerr and Frese put forward an explanation for the 20% to 30% completion rate they found among students: non-completion was attributed to a combination of being unprepared, not motivated, short on time, or simply not appreciating the value of the reading. Three separate studies by St Clair-Thompson, Graham and Marsham (2018) showed that university expectations exceed what students are prepared to put into their academic reading. Focus groups identified the reasons as being as much about course structure and a sense of limited time as about any perceived benefits. Then there is the paradox laid bare by Gorzycki, Desa, Howard and Allen (2020): undergraduates will tell you the reading has merit, but their actions say otherwise, viewing it as irrelevant to what transpires in class and more than what is required.

For the system's design, this has two consequences. An intervention needs to ensure that genuine engagement is more rewarding than going around it, and that reading matters. And one should be wary of a punitive approach to verification; with the tools at hand today, the student can have it his way in, making strict enforcement unreliable.

Social Annotation

With social annotation systems, a class can read a shared document together and add their own highlights, queries, and discussion around particular passages. In a 2020 review of research on online classes, Zhu, Chen, Avadhanam, Shui, and Zhang make the

case for the value of such systems in fostering comprehension and facilitating connections among learners. The benefits are also evident from the findings of Miller, Lukoff, King, and Mazur (2018): an introductory physics course that had students use a social annotation tool for pre-class work saw not only high compliance but also quantifiable engagement with the material. Then there is the matter of how one views annotation; Kalir and Garcia (2021) would have it as a form of social knowledge construction as opposed to something done in private, a perspective this project is set on following in that it makes of the student's solitary exercise a social act for all to see. And to overcome the self-consciousness that is a known impediment to participation in any classroom questioning, Assessly has an anonymity feature so the instructor or a peer cannot tell who the author is.

Retrieval Practice and Active Recall

In the field of cognitive psychology, there is no more solid finding than the testing effect, which holds that the act of pulling information from one's memory is a better way to commit it to retention than restudying. This was borne out in a 2017 meta-analysis of the practice-testing literature by Adesope, Trevisan, and Sundararajan, who found practice tests to be superior to restudying and other conditions. A systematic review of fifty classroom experiments conducted by Agarwal, Nunes, and Blunt (2021) likewise found medium-to-large gains from retrieval practice for the most part, regardless of the test format or the level of education and content involved. With its flashcard mode, Assessly puts these conclusions to work. The student is required to type an answer from memory to make the card turn; a language model then scores the entry based on the reference answer provided by the card. To ensure the retrieval attempt is authentic, pasting into the answer field is not permitted.

Artificial Intelligence in Reading and Assessment

Automatic Question Generation

A systematic review by Kurdi, Leo, Parsia, Sattler and Al-Emari (2020) on automatic question generation in an educational setting put a finger on some enduring problems with trivial surface questions and the need to ensure they are answerable and of controlled difficulty. One finds the same challenges at work on this project; as detailed in Chapter 5, it was necessary to put in place prompt-level content rules to prevent early quizzes that would ask about document metadata, such as how long one might take to read it. The advent of large language models has put an end to such concerns in many respects, Kasneci et al. (2023) note the gains in topical relevance and fluency, and per-document question pools can be assembled when the document is uploaded without issue.

Automated Scoring of Open Responses

The research behind automated essay scoring is extensive, as Ke and Ng (2019) have documented, and current systems can match human raters on more limited tasks. In the case of Assessly, language models are put to use on two such bounded exercises: the key-point analysis of a short essay for partial credit and feedback, and the 0-3 evaluation of a typed flashcard in relation to the reference. Being of a narrower scope than open essay scoring, the score is given with the reference material so the student may see the reasoning behind it. It is a form of transparency in keeping with what the feedback literature would have (Wisniewski, Zierer & Hattie, 2020).

Dialogue Tutoring

A meta-analysis by Kulik and Fletcher (2016) of some fifty controlled studies on intelligent tutoring systems found a median advantage over more traditional instruction

of 0.66 standard deviations. Wollny et al., in their 2021 review of educational chatbots, saw much to recommend in conversational agents but observed that few were designed to avoid putting words in the student's mouth. That is what lies at the heart of how Assessly has put its tutor together: a server-side guard and system prompt ensure that the tutor will not offer conclusions but will only ask questions to guide the learner through his own reasoning, as one would in the Socratic manner. Were the tutor to give an answer outright, it would be no more than a means of getting around the reading.

Feedback and Formative Assessment

In a meta-analysis of 435 pieces of research on the subject, Wisniewski, Zierer and Hattie (2020) report a medium effect on learning from educational feedback in general, though they make clear that the more information the feedback contains, the better: elaborated feedback with a focus on task and process leaves bare grades or praise well behind. The grading design at Assessly is built on such findings. The composite grade is weighted for process, and each part for which there is no justification has a specific action attached; the student can see the grade breakdown while he is still at work. In this fashion, grade transparency serves as a form of feedback, not a final word.

LMS Integration Standards

The platform is built to the Learning Tools Interoperability (LTI) 1.3 specification, which relies on OpenID Connect for tool launches and signed JSON Web Tokens (JWTs) to add a security layer to launch messages (IMS Global Learning Consortium, 2019b). For any authorized external tools wishing to report scores to the LMS gradebook, Assessly follows the Assignment and Grade Services specification (IMS Global Learning Consortium, 2019a).

Compliance with these standards enables the platform to be readily adopted by an institution's existing systems. A launch via the LMS means there is no requirement for users to go through a separate login; the user and course context will ensure the launch is tied to the right learning environment. The composite mark from the platform can be passed back and entered into the gradebook. To maintain a firm link between the platform's assignments and LMS activities, an additional binding layer has been implemented as part of this project (see Chapter 3).

Theoretical Framework

The evaluation is put together and made sense of through a pair of frameworks. On the one hand, the ICAP framework accounts for the importance of the engagement behaviors that the Assessly instruments elicit; on the other, self-regulated learning theory explains how the platform will use the signals it receives.

The ICAP Framework

According to ICAP theory, learning activities can be put in one of four modes of cognitive engagement: Passive, Active, Constructive, or Interactive. Chi et al. (2018) have presented evidence showing that as students advance along this spectrum, their learning improves. In a single reading workflow, Assessly puts the entire spectrum into practice. A student is passive in the standard viewer with the document, but active when he or she highlights. To be constructive, one has to put forth something not already in the text, as is the case with composing an essay, putting up an annotation question, or typing out flashcard answers from memory. There is interaction through conversation with the Socratic tutor or by replying to a classmate's annotation. With 60 of the 100 points in the composite grade going to these latter two types of behavior, the grading scheme is not an

arbitrary rubric but a commitment informed by theory, compensating students for the very modes of engagement that ICAP would consider most generative.

Self-Regulated Learning

Under the model of self-regulated learning, good studying is a matter of forethought, performance and then self-reflection, all underpinned by feedback and self-monitoring. Panadero (2017), in his review of the principal cyclical models, including Zimmerman's, makes much of the external supports used to scaffold the process. Assessly takes each of these phases and externalizes it. A student can act on what the live grade indicator shows as having been earned during performance. As for reflection, the next-step recommendation surface does some of the work for the student, turning the record of his or her behavior into a suggestion for action. At the class level, the instructor dashboard serves to close the loop. And the framework would have it that study behavior is shaped by feedback on the process, which is borne out by the pilot, where the majority of participants were found to be after the engagement components (see Chapter 5).

CHAPTER 3

SYSTEM DESIGN

Design Principles and Requirements

The design is underpinned by three tenets. The first is to put engagement ahead of verification; integrity should not be adversarial but in proportion to the task, while each feature is made as rewarding and straightforward as possible for the user. We also adhere to a signals-over-models approach: any personalization on the surface must come from an SQL aggregation of the transactional tables, where one would find records of normal user activity, eschewing a dedicated learner-modeling service and the attendant costs of schema duplication and stale data. Lastly, the platform is to be embedded in the institution's current Moodle workflow via LTI 1.3, not exist alongside it.

As for what the platform has to do, it must allow instructors to upload PDF readings to courses and create flashcard decks and question pools from the text. Students are to be given a Socratic tutor and a reader operating in standard, annotate, or flashcard mode. The system will compute and display a running composite grade, provide a class dashboard, and offer individual-level recommendations. Quizzes are to be run with soft lockdown and per-student time overrides, sampling questions on each attempt. On the non-functional side, there are requirements for privacy in anonymous annotation, some flexibility with free-tier infrastructure, and resilience in the event that an LLM provider goes down.

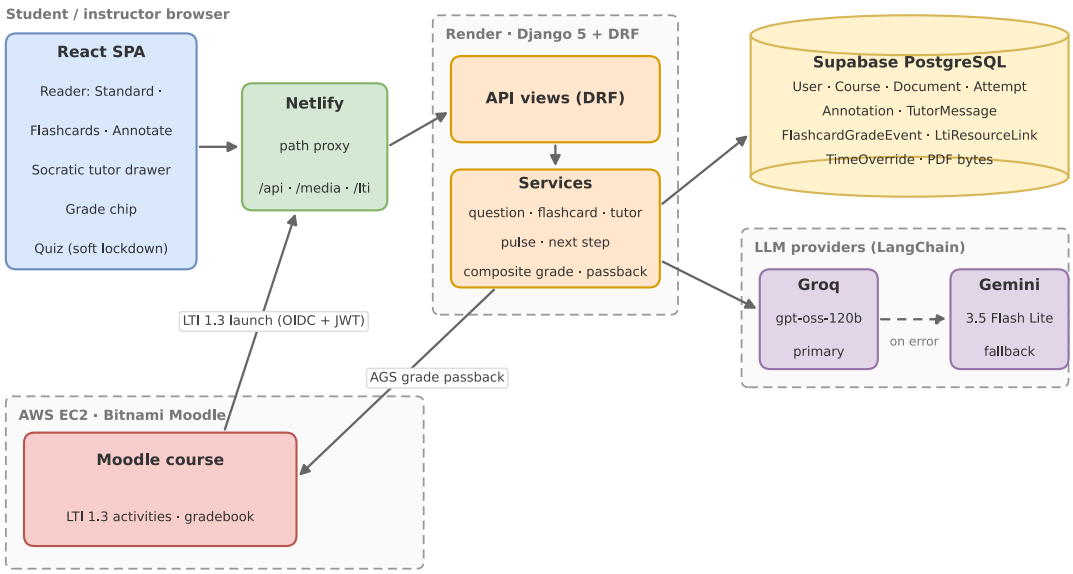
Architecture

The architecture of the Assessly platform is a two-tier web application (see Figure 1). It comprises a React single-page frontend and a Django REST Framework backed by

PostgreSQL for record-keeping. For production use, the CDN host will serve the frontend and set up proxies for any API, LTI, or media paths to the backend; this is done so the session cookie comes from the frontend origin, as the cross-site LTI launch flow dictates.

On the backend, domain logic is handled by service modules that cover everything from PDF extraction and question or flashcard generation to grading, Socratic tutoring, recommendations, instructor pulse, composite grading, and grade passback; thin API views sit on top of these. A Moodle instance running on a cloud VM serves as the demo LMS and has been registered with the platform as an LTI 1.3 tool.

Figure 1
System Architecture of the Assessly Platform



Data Model

There is no distinction between the transactional and analytics models; the former serves for both. The model's core entities are User (role, LTI identity), Course, Document (the reading, complete with any extracted text as well as a cached pool of questions and flashcard deck), Attempt, Question, and Answer. On top of this, the engagement layer

has Annotation (a question or comment, both carrying an anonymity flag) and AnnotationComment to handle threaded replies, plus TutorMessage for the Socratic chat log and FlashcardGradeEvent to record the score given by the AI for each typed attempt at a card. As for integration, one finds LtiResourceLink to establish a lasting connection between an LMS activity and a document while preserving the gradebook column reference, and TimeOverride to manage student quiz time allowances. All grading and personalization done by the platform amount to queries against these tables, with no separate store for a derived learner model.

The Engagement Layer

Social Annotation

When in annotate mode, the PDF is presented to the reader with a selection layer. Highlighting any text will bring up an inline composer for a comment or question, which can be left anonymous. Class annotations are kept in a sidebar complete with threaded replies and kind filters; these are updated via polling. On the server side, we ensure anonymity from both the instructor and peers. An author's name will appear on their own post as "anonymous" for the purpose of management, but no one else will see it. As for the instructor, a heatmap of annotation density by page will highlight the most discussed passages while keeping the annotator's identity anonymous. Each contribution receives a private AI quality upon submitting an annotation like comment, question, or replies, and score visible only to its author and the instructor. Also replies can address an earlier reply through a mention tag by first name.

Active-Recall Flashcards

Uploading the document creates the flashcard deck. One does not get to flip a card until an attempt has been typed in (pasting is not allowed) in response to the task set out

on the front, be it a directive to explain or define or a straight question. The language model then scores the effort on a 0-3 scale and provides feedback relative to the reference answer; the server records each graded attempt and its card identity to determine mastery at the deck level. Any local self-assessment, such as flagging a card for review or indicating it is understood, is handled client-side.

The Socratic Tutor

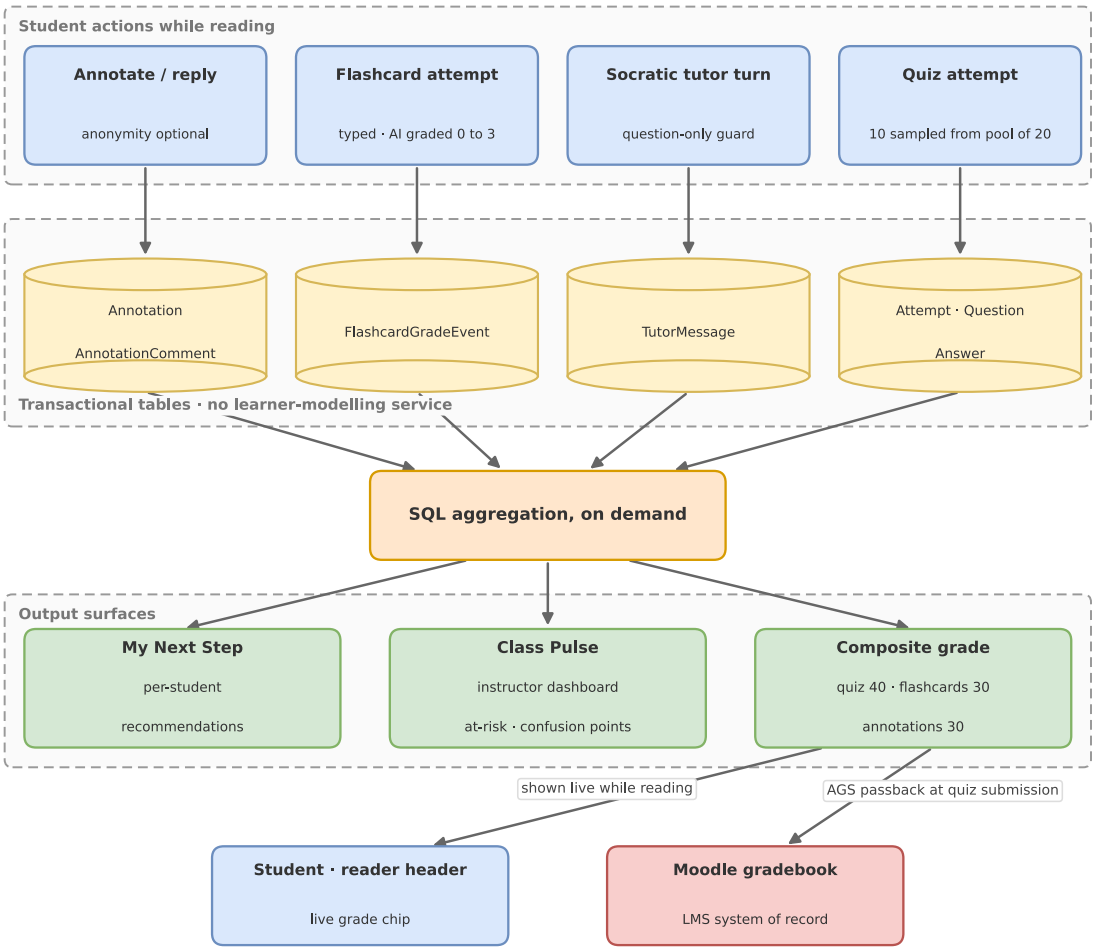
The tutor is a per-document chat drawer available while reading. Its system prompt instructs it to guide with questions grounded in the reading and never to hand over answers, and a server-side guard enforces the contract by checking that every response ends as a question, regenerating otherwise. The tutor is deliberately absent from the assessment environment.

Cross-Signal Personalization

The behavioral signals are used by two surfaces: one for the student and the other for the instructor; Figure 2 shows how they are fed from the transactional tables. On the student side, My Next Step will collate flashcard state with any annotation activity or quiz results and put together a prioritized list of what to do next, be it to reread the pages on which questions were posed or to work on cards not yet in hand. For the instructor, there is Class Pulse, which brings these same signals to bear at the class level: the annotation passages most talked about, where quiz accuracy has been poorest, and those students whose profile (a low score and many open questions, or no attempt at all) puts them at risk. SQL aggregation performs the necessary computation for both, so there is no need for a background pipeline.

Figure 2

Cross-Signal Data Flow From the Transactional Tables to the My Next Step and Class Pulse Surfaces



Composite Engagement Grading

A reading assignment's grade is made up of three elements. The first is the student's top quiz score, scaled to 40 points. Then there are 30 points for flashcard mastery, which a student puts on record if his or her best graded effort on each card adds up to half the deck's marks. The remaining 30 goes to annotation. Here the focus is on annotation quality, not just participation. Before an AI is even called upon, a free heuristic

gate will have weeded out any near-empty or nonsensical text; what is left is then put to the test by the language model, which scores every reply and annotation from 0 to 2 in relation to the student's highlighted passage and the document in discussion. To be awarded this component, a student must have put in at least three comments or replies deemed on-topic and gained four quality points. In the case of questions, only the one with the highest score is tallied, precluding any attempt to make up for a lack of real discussion with filler or repetition. The instructor and the author of the annotator can view the score, but it is not visible to peers. This ensures integrity and privacy for grades among students.

AI Service Design

Generation and grading are handled by a single LLM models across all flows. The client uses a Groq-hosted open-weight model as the primary provider and a Gemini model as the fallback when an error occurs. To keep within budget, the document text is cut short, with a bias toward the opening and closing of the piece. There are also explicit content rules in the generation prompts. Questions and cards should focus on substance, not publisher boilerplate, front matter, or document metadata; a task must be spelled out on the face of every flashcard. These rules were strengthened after failures observed during the field pilot, as detailed in Chapter 5.

Temperatures are set according to the task, with grading using the lowest settings. Annotation quality scoring adds a sixth task to the service and runs at temperature zero. Each annotation or reply is scored from 0 to 2 against the highlighted passage. A heuristic check first assigns zero to near-empty or nonsense text without making an LLM call.

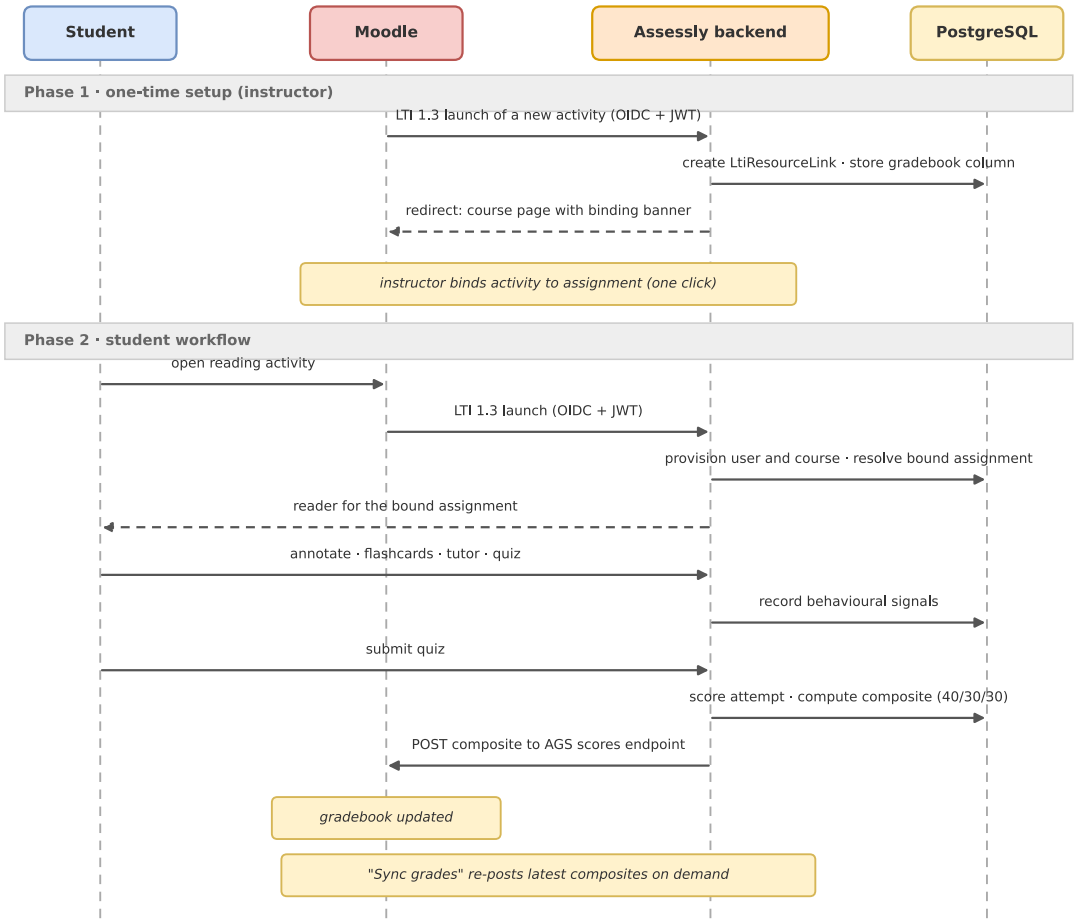
Contributions or annotations that pass this check are assessed through the same primary and fallback providers used by the other flows.

LMS Integration and Grade Passback

With Assessly, LTI 1.3 launch is done via OIDC initiation. Users and courses are provisioned from the launch's claims under a promote-only role policy; the full sequence of launch, binding, and passback is shown in Figure 3. An instructor needs only one click to bind an LMS activity, which is put down as a resource link on its first launch, to an assignment on the platform. From then on, the activity will open directly to the reader for that assignment, and the gradebook column will be retained. The composite grade is posted by way of the Assignment and Grade Services scores endpoint; the bound resource link takes care of resolving the column, so the gradebook destination is not a function of how one navigates. There is a resync sweep available to the instructor for retries, but nothing is posted for accounts lacking an LTI identity, which precludes the possibility of crediting the wrong user in the LMS.

Figure 3

LTI 1.3 Launch, Activity Binding, and Grade Passback Sequence



Assessment Integrity and Accessibility

The integrity measures are commensurate with what is called for in a formative reading setting. Ten questions are drawn at random from a pool of twenty for the document in question for every attempt, meaning any answers put around will be of less worth to the class and on subsequent tries. The quiz is under a soft lockdown of a kind: we ask for full screen and detect when a tab is switched, or the window is blurred, or full screen is left, and copy and paste are off limits. Any infraction is entered in an auditable record for the attempt, and three will trigger an automatic submission. It is deterrence and

audit, not prevention, and students are told as much. As for accessibility, time is differentiated; the server will apply an override for an assignment at the start of an attempt if the instructor has set one for a particular student.

CHAPTER 4

IMPLEMENTATION

Backend

Django 5 and Django REST Framework form the basis of the backend. To keep HTTP handling and domain logic thin and testable, every API resource is an explicit view class built on top of a service module. In addition to the usual CRUD for resources, there are a number of endpoints of interest: the per-student live-grade endpoint that drives the reader's indicator; instructor endpoints to resync grades or bind LMS activities; the time-override roster; and the aggregation endpoints for pulse and next-step. The annotation and tutor endpoints have rate limiting in scope, but it applies only to writes so as not to let the sidebar's polling reads consume a student's quota, something production has made clear is needed. When a PDF is uploaded, PyMuPDF extracts the text and chunks it for downstream AI; the bytes are stored in PostgreSQL, so they persist even if the application filesystem is ephemeral.

Frontend

A React single-page application makes up the frontend. Its token-based design system means that one theme module is the source for all color, elevation, and spacing, while the reading surfaces share a common card idiom. On the document page, the user will find three reader modes: annotate mode, which brings in the heatmap and annotation sidebar; a flashcard player for typed recall; the tutor drawer; and a grade chip in the header. This last is persistent and can be expanded to a popover showing a roadmap of progress (quiz, flashcards, and annotations) with live hints. The quiz page has client-side listeners for its lockdown and will submit any violation log along with the attempt. As for

the course page, it is home to the management surfaces for the instructor (analytics, time overrides, grade resync, and activity binding) and a static explainer of how grades are distributed to students.

Deployment

In production, the setup is as follows: a CDN host with path proxying for the frontend, a managed container host for the backend, and a cloud VM running the demo Moodle, with PostgreSQL via a session pooler on a managed database service. The pilot turned up two deployment defects worth noting. One was the application host's ephemeral filesystem, which would ensure that uploaded PDFs were gone after each deployment, necessitating that the document bytes be stored in the database. Then there was the web server on the Moodle host that had a habit of removing the Authorization header from service requests before they reached PHP, so authentication for a gradebook write would not hold until we put the header back at the web server level. The defect log in Chapter 5 records these.

Privacy and Security

The serialization layer will not allow an annotation to be anything but anonymous; the name and user identifier are off-limits to any viewer who is not the author, whether peer or instructor, though the database retains the identity for abuse handling. For cross-site sessions, we rely on SameSite=None cookies with the secure flag over the proxy and JWT validation of LTI messages, with CORS origins kept strict to compensate for the CSRF exemption. A grade passback will post only for LTI-linked accounts and will sign its token requests with the key pair registered to the platform.

CHAPTER 5

TESTING AND EVALUATION

Automated Testing

An in-memory database is the environment for over 131 tests that make up the automated backend suite. These tests must pass before any deployment. The scope of these tests is cuts across: scoring logic, flashcard generation and its grading, the write-only throttle regression, and the annotation API with its anonymity masking. There is also testing of the Socratic tutor's question-only guard, LTI resource-link binding as well as line-item resolution, and the pilot telemetry export. We run the suite against composite grading at every threshold edge, per-attempt question sampling, time overrides, and media persistence in a simulated redeploy. Instructor pulse and next-step aggregations are covered, along with the grade passback and resync behavior using mocked responses from the LMS.

Field Pilot with Real Users

The platform was tested in a formative field pilot on the live production stack. The object of the pilot was to see how well the central infrastructure held up and to gauge feature adoption, integration stability, and usability. In particular, we wanted to know whether the personalization and grading designs would be driven by behavioral signals from real users who had not been instructed to provide them, as they used the four channels. Learning outcomes were not part of the equation; without a pre- or post- test or a comparison group, there was no way to measure them.

Method

Participants were drawn from six higher-education institutions and enrolled as students in a pilot course with a single reading module; in all, 11 students put themselves forward. Over the course of 14 days, they were left to their own devices to get through the entire workflow of the pilot, from the timed quiz to annotation and flashcards, the tutor, and the reading itself. With informed consent obtained from all participants, the data are presented anonymously (P1 to P11 for telemetry and R1 to R9 for surveys).

Three data sources were combined for the report. One was anonymized system telemetry exported by a custom management command. Another was a defect log in which every production defect was accounted for down to the fix commit and root cause. The third was a survey, returned by nine of the participants (82%), that included a retention question, usefulness ratings for each feature, and a three-part usability scale modeled on Lewis' System Usability Scale (2018). As only three of the ten SUS items were used, Assessly makes no claim to an SUS score or to any benchmarking. The protocol had originally called for two reading modules but later saw it fit to drop the second to allow greater depth in one; this meant all participants had the same content but precluded the system from gathering evidence on repeat usage.

Engagement and Channel Adoption

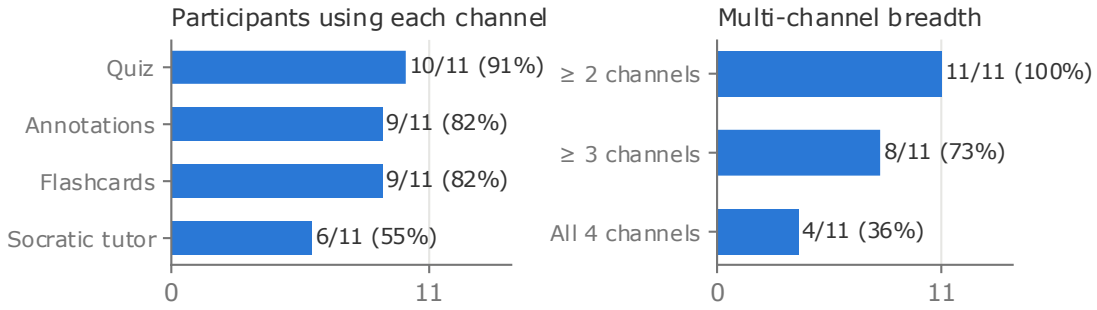
As shown in Figure 4, there was a fair degree of channel adoption: all 11 participants used at least 2 of the 4 signal channels. A majority (73%) used 3 or more, while for 4 (36%) used all of them.

Quiz results tell a similar story of engagement. 10 of the 11 put in at least an attempt; the best scores ranged from 40 to 100% (a mean of 73 and a median of 75). The

policy of taking the best attempt had its effect when one of the three who retook the quiz saw his counted best score rise to 70%, compared with an attempt mean of 45.

The deterrence-with-audit approach yielded audit data of a genuine sort, evidenced by the 4 lockdown violations and the automatic submission that followed a third violation from one participant. As for annotations, 22% of the 18 (from 4 separate participants) were submitted under the anonymity option.

Figure 4
Signal-Channel Adoption During the Field Pilot (n = 11)



Composite Grades and the Cross-Signal Claim

The median for composite grades was 58 on a 0-100 scale. As shown in Figure 5, each participant's grade is divided into its annotation, flashcard, and quiz components. Only 5 of the 11 made the grade for the flashcard component, and six for the annotation component. There was no ambiguity at the boundary with the flashcard threshold; those who earned it put up deck mastery ratios of 0.53 to 1.0 as against the 0.11 to 0.25 of the rest.

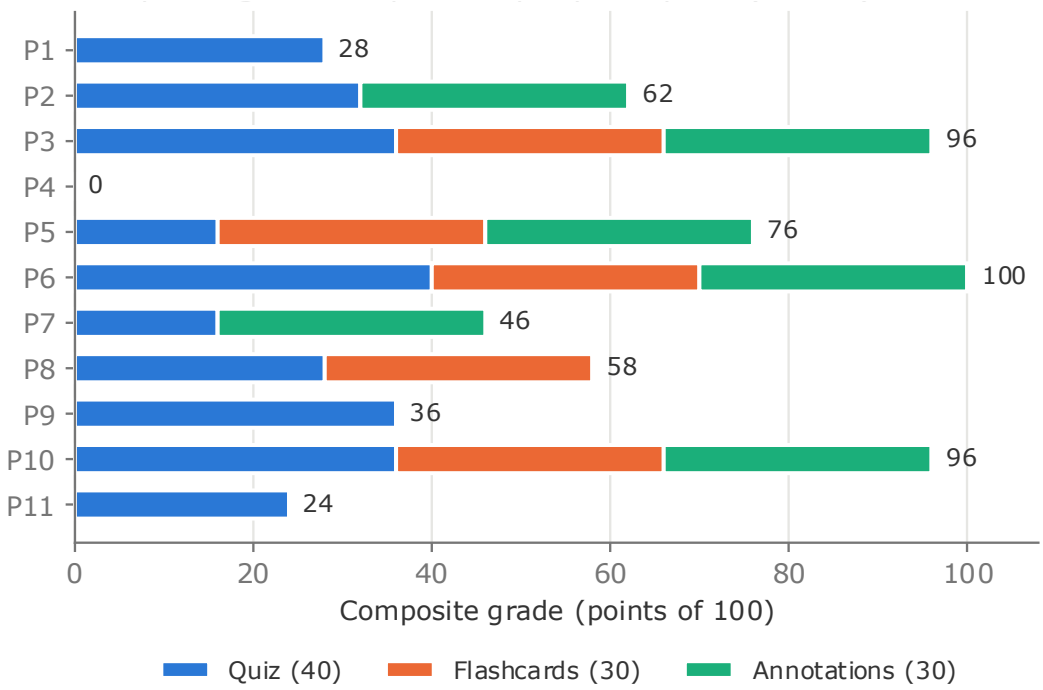
One need only look at P9 and P10 to see what the composite is about. Each put in 90 % on the quiz. Yet P9 did not use any other channel, and his composite came to 36.

P10, by contrast, annotated on five occasions and completed eleven flashcards for a 96. Then there is P5: a 40 per cent on the quiz was offset by considerable activity with annotations and flashcards to bring the total to 76. Put in ICAP language, the composite makes distinctions between engagement profiles that would be indistinguishable under a recall-only grade.

In this way the claim of the thesis is borne out in the infrastructure. The system has drawn its recommendations, the class dashboard, and the composite for the LMS gradebook from SQL aggregations of the signals that real users have put out across all four channels, obviating the need for an independent learner-modeling service.

Figure 5

Composite Grade Composition Per Participant (N = 11)



Survey Results

For the 9 items in the adapted usability set, the mean was 4.00 on a five-point scale for wanting to continue using the system; ease of use averaged 4.22, and complexity

was rated low (reverse-scored mean of 4.56). The composite for these three is 4.26 (SD 0.81). As shown in Table 5.1, no feature was rated below 4.0 for usefulness.

When it comes to retention in an actual course setting, 7 of the 9 would continue with Assessly, one was noncommittal, while another's use would hinge on the AI having a wider scope and better response times (Figure 6). A case in point is the Socratic tutor, which, by itself, was the least well-received in both the survey and telemetry data: 55 % adoption and a 4.00 usefulness rating. This convergence points to discoverability and to how the tutor formats its responses as areas where improvements are most needed.

Table 1

Per-Feature Usefulness Ratings (n = 9, scale 1 to 5)

Feature	**Mean**	**SD**	**Rated 4 or higher**
Timed quiz	4.89	0.33	9/9
Annotations	4.89	0.33	9/9
My Next Step	4.78	0.44	9/9
Grade tab	4.78	0.44	9/9
Flashcards	4.67	0.50	9/9
Socratic tutor	4.00	0.87	6/9

Figure 6

Per-Feature Usefulness Ratings from the Final Survey(N=9)

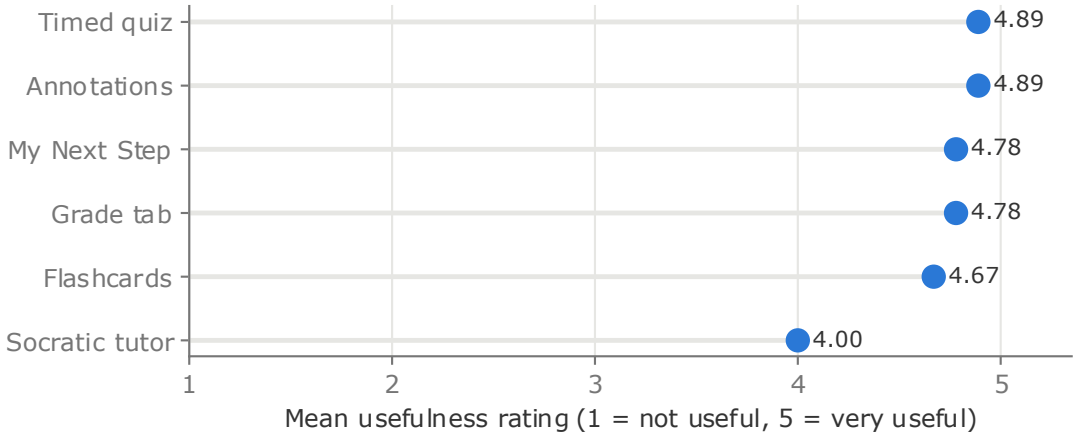
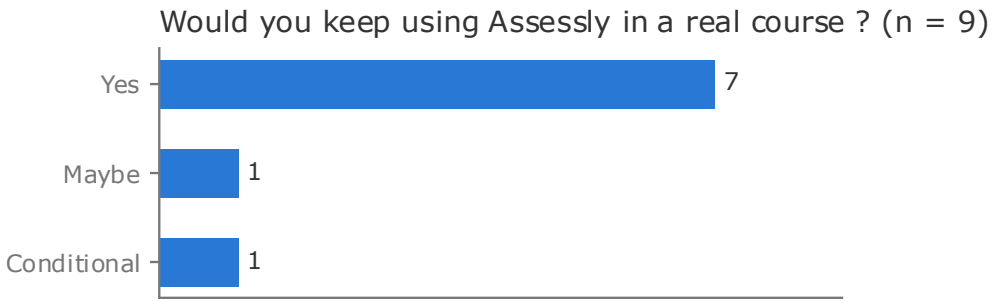


Figure 7

Retention Intent Among Survey Respondents (n = 9)



Qualitative Findings

There was not an abundance of free-text responses, but what there were carried weight. The design for anonymity was directly approved of; as one put it, the ability to make anonymous annotations "encourages participation without making students feel self-conscious" (R2), which is precisely what concealing one's identity from an instructor or peer is meant to do. That respondent also pointed out two usability issues: that a colored passage is difficult to re-select for re-annotation, and that the lockdown allows window switching. One could call the latter a failure of the system was it not by design; the

lockdown is there to log departures, not forestall them, and this respondent's violation log bears out that detection is in place. Still, reading the logs as non-enforcement is, in itself, a finding in terms of communication.

As for other points, R8 noted an absence of any self-service way to recover an account after a password was lost. R2 made the case for more in the way of formatting in a tutor's reply, while R8 wanted the tutor to go beyond the reading for sources, at odds with the reading-centric design on purpose. There was a plain aesthetic objection from R7. And R9 had no trouble saying the system was preferable to what their institution has at present.

Defects Found and Fixed During the Pilot

Because the pilot was conducted on the live production stack, it also served to put the deployment through its paces. For the seven defects that came to light during the pilot period, a root cause could be ascertained from the project repositories, and a fix was put in place (see Table 2).

Table 2

Production Defects Found and Fixed During the Pilot

Defect (as observed)	Root cause	Fix
Annotation posts rejected with 429 for active readers	Sidebar list polling shared the creation rate-limit scope	Throttle applied to writes only (6abf35c)
Quiz grades posted to the wrong Moodle column, or none	No durable binding between LMS activity and assignment	Resource-link binding model (c39ce46)
Grade lost when the LMS timed out; no retry existed	Ten-second HTTP timeout, single-shot passback	Longer timeout, posted-state tracking, resync sweep (a220409)

Defect (as observed)	Root cause	Fix
All gradebook posts rejected as unauthorised despite valid tokens	Moodle host's web server stripped the Authorization header	Header pass-through in the Apache virtual hosts (server config)
Uploaded PDFs disappeared after every deployment	Ephemeral application filesystem on the hosting tier	Document bytes persisted in PostgreSQL (de90ce0)
Quiz asked about document metadata such as estimated reading time	Extracted text includes front matter; prompts had no content rules	Substance-only generation rules (d7bd868)
Flashcard fronts appeared as bare statements with no task	Generation prompt permitted statement-style fronts	Fronts must pose a question or explicit task (2926ec7)

Threats to Validity

As is to be anticipated in a formative pilot, the evaluation has its limitations. With a self-selected, modest-sized sample (11 for telemetry, 9 for the survey), there is an element of acquiescence bias to contend with; recruitment from the researcher's network would seem to account for the feature ratings clustering ter in the 4.0 to 4.9 range. The fact that the researcher also developed the work is not fully dispelled by the passive observation made during the reading window, though it is mitigated. Nor can one tell novelty from sustained engagement in 14 days with but one module, let alone have any indication of repeat usage. What the three-item adaptation of the usability instrument yields is indicative, not something to be benchmarked. On the question of composite weights and thresholds, these are matters of pedagogical judgment underpinned by the ICAP framework, rather than being determined empirically. In one instance, a survey respondent was internally at odds, registering strong disagreement with frequent use while, in writing, he preferred the system to what is currently available and meant to continue with it; this points to a misreading of the scale, and the response has been left as

is. Then there is the platform itself, which, with defects being remedied, was not the same software for the early and late comers in the pilot; the defect log will bear that out.

Evaluation Against Objectives

The 7 objectives set out in Section 1.3 have been met in full. Real users used the engagement layer (Objective 1), which was implemented with three features; both the annotation and flashcard channels saw an 82 % take-up. For cross-signal personalization (Objective 2), the pilot ran on SQL aggregation only. Every submission resulted in a composite grade (Objective 3) being tallied and sent to the Moodle gradebook. As for the AI pipeline (Objective 4), it assembled the assets and decks for the module, and content rules were made more robust where failures were noted. Once the defects were corrected, LTI integration with binding and dependable passback (Objective 5) could handle actual traffic. The measures for integrity and accessibility (Objective 6) yielded proper audit data and time overrides for individual students. Finally, the evaluation (Objective 7) has drawn on over 131 automated tests and the field pilot described above.

CHAPTER 6

CONCLUSION AND FUTURE WORK

Summary

Rather than policing the results of reading to stop non-compliance, the project has taken the approach of instrumenting and socializing the reading process, making it something to be engaging and rewarding. What one finds on the platform is an engagement layer for PDF reading, featuring Socratic tutoring, active-recall flashcards, and social annotation. Behavioral data from these are collated via unadorned SQL into a class dashboard and into individual recommendations for each student. The grade is calculated with engagement accounting for most of the marks, and LTI 1.3 is used to provide reliable grade passback so that all of this is handled within the institution's Moodle workflow. An end-to-end field pilot involving 11 users from as many as 6 institutions has shown that the approach holds up in actual use.

Contributions

This project makes four contributions to the design and engineering of reading engagement systems for higher education.

First, it demonstrates that cross-signal personalization can be achieved in the absence of a dedicated learner-modeling service. The system's per-student recommendations, composite grades and class-level instructor dashboard are all obtained by way of SQL aggregation from the transactional tables where students actions are recorded. A field pilot put the architecture to the test in actual use. Uninstructed users supplied the aggregations with true multi-channel input. 73 % on 3 or more, and 36% on all 4 emitted behavioral signals of the 4 available channels.

Second, it contributes a theory-informed composite grading design that puts the ICAP framework to work in an LMS gradebook. The scheme is set up so that 60 of 100 points are accounted for by what the framework deems the most generative forms of engagement, such as annotations or trying one's hand at flashcards. The current annotation component also requires AI-scored contribution quality. Students receive its 30 points only after accumulating at least four quality points and at least three comments or replies scoring 1 or higher. Only the student's best-scoring question counts toward the quality-point total. These conditions reward substance rather than volume. In the course of the pilot, the design was found to separate profiles of engagement that would otherwise be indistinguishable under a system of recall-only grading. One participant might have a 90 % on the quiz and end up with a composite of 36, another 96; yet a third, with just 40 % on the quiz, could make 76 from his engagement with the reading.

Third, it contributes a Socratic tutoring approach designed help students in understanding the reading and to reduce the risk of students using the AI assistant to bypass the reading. To ensure the tutor is guided by text-based questions, a system prompt is in place, and a server-side guard ensures that any response that does not end in a question is regenerated. These are constraints the automated tests cover, and the pilot upheld.

Fourth, it contributes an LTI 1.3 integration pattern in which LMS activities are durably bound to tool assignments through a persistent resource-link model, making grade passback independent of the path by which a student reaches an assessment.

Limitations

One can place the limitations of the work in three categories: how the evaluation was done, the design calibration, and the extent of the security and privacy guarantees.

As a formative exercise, the evaluation has shown the platform to be usable with its cross-signal infrastructure operating end to end in production, and that real users will make use of the engagement features on their own. But it is not in a position to put forward causal claims as to any effect on learning outcomes or reading compliance; one cannot do that with a formative pilot of 11 self-selected individuals and a single reading module and no group for comparison. The survey results are more indicative than generalizable given the sample size.

Three guarantees have been put in place with deliberate bounds. While the system has structured prompts and content rules to stave off the more obvious failures, the AI questions, feedback and flashcards are still open to the kind of hallucination large language models are prone to; the best safeguard is for the instructor to review the material prior to publication. The lockdown environment is more of a deterrent than a prevention, in that it will detect and record any leaving of the assessment context but does not preclude it. The platform is forthright with students about this stance, though a participant in the pilot was inclined to see it as a lack of enforcement.

Future Work

Controlled Efficacy Study

This project was scoped to avoid being the very thing that should come next, namely a controlled evaluation with pre- and post-measures and comparison groups to determine if the cross-signal personalization and the grading by engagement are in fact conducive to better comprehension and reading compliance. Two designs lend themselves to it. With a between-groups approach, one could pit sections running Assessly against those with conventional quizzes and measure compliance in behavioral terms.

Tutor Improvements

Three enhancements to the tutor are called for by what the pilot had to show: more elaborate formatting of responses, easier discovery within the reader and, for the tutoring of lengthier documents without breaking the question-only contract, retrieval-augmented grounding from an index of the document text.

Support for Additional Document Types and Video

With further development, Assessly can be made to do more than handle PDF readings and accommodate Word files, presentations and video. Any document support would have to involve text extraction in a way that keeps the original format's structure intact. In the case of video, transcripts and timestamps would be used to tie annotations, quizzes, flashcards and tutor questions to particular segments. Such an expansion would put the platform in a position to foster engagement with a broader array of course content.

APPENDIX

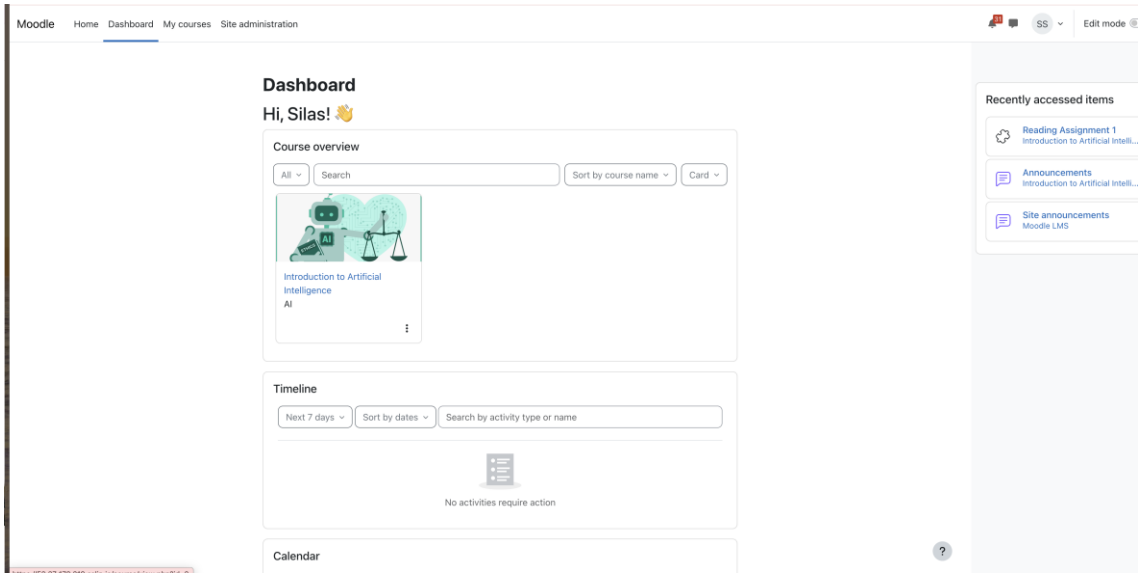
Assessly Links

Moodle:

https://52.37.172.219.sslip.io/login/index.php

Username: silas

Password: User@123456



Assessly direct access without Moodle route:

https://assessly-front.netlify.app/

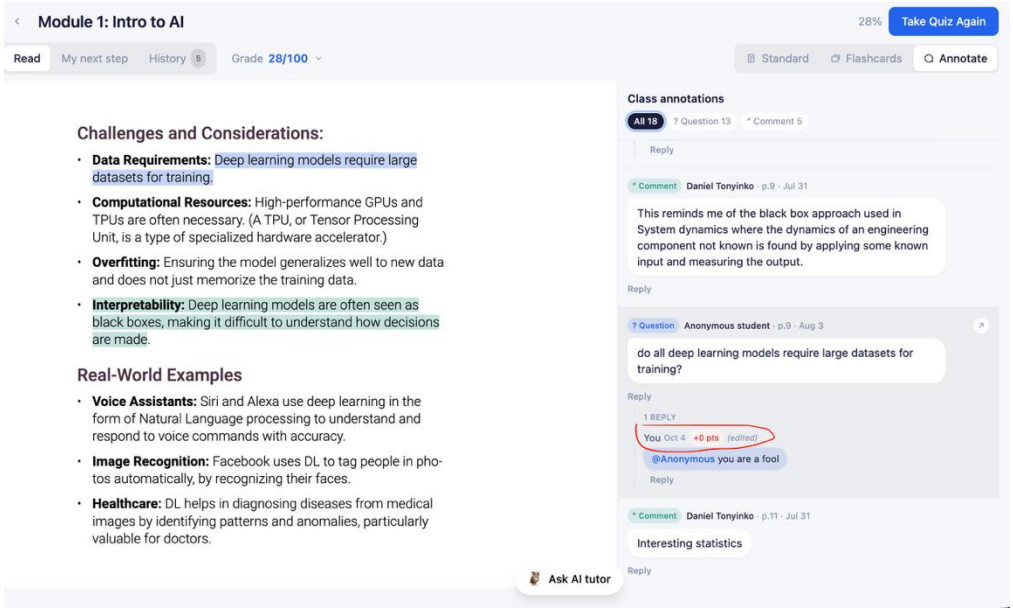
Username: silas

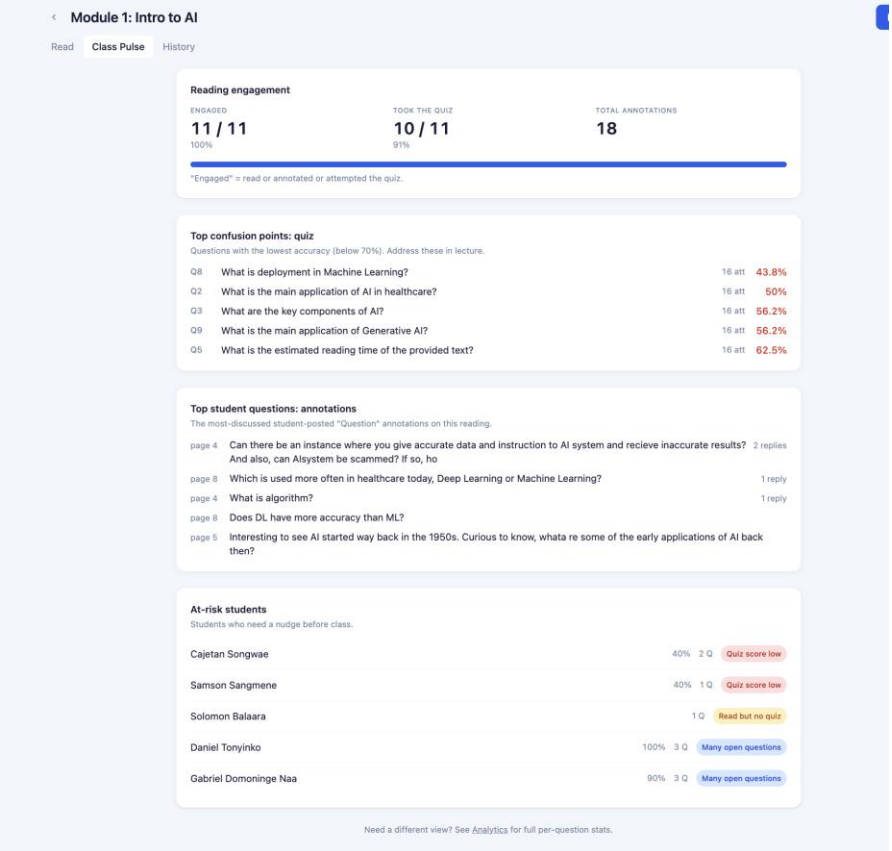
Password: User@123456

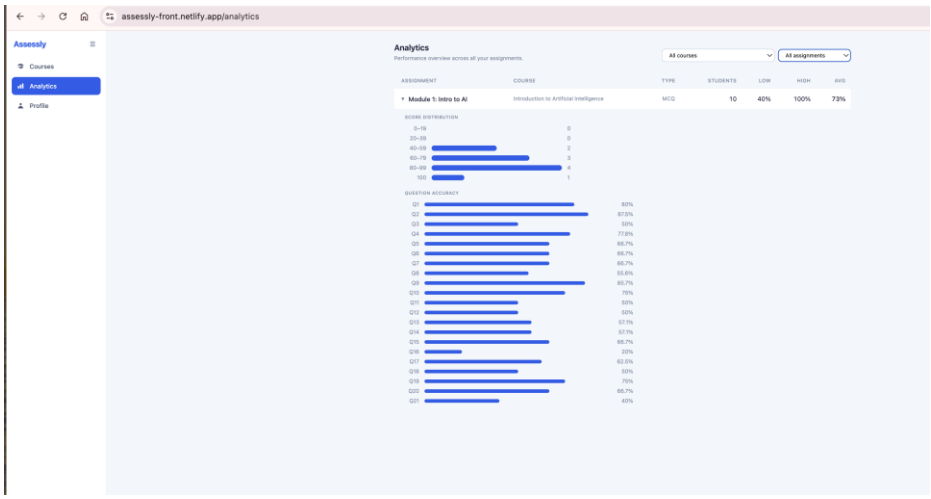
Survey Questions and responses

https://docs.google.com/forms/d/e/1FAIpQLSdodz4SOabuX6H6_MbS5Oqaofdm6YDeKlRTaqQ8R08GOmD3Tg/viewform?usp=sharing&ouid=103840876915912619978

System Screenshots

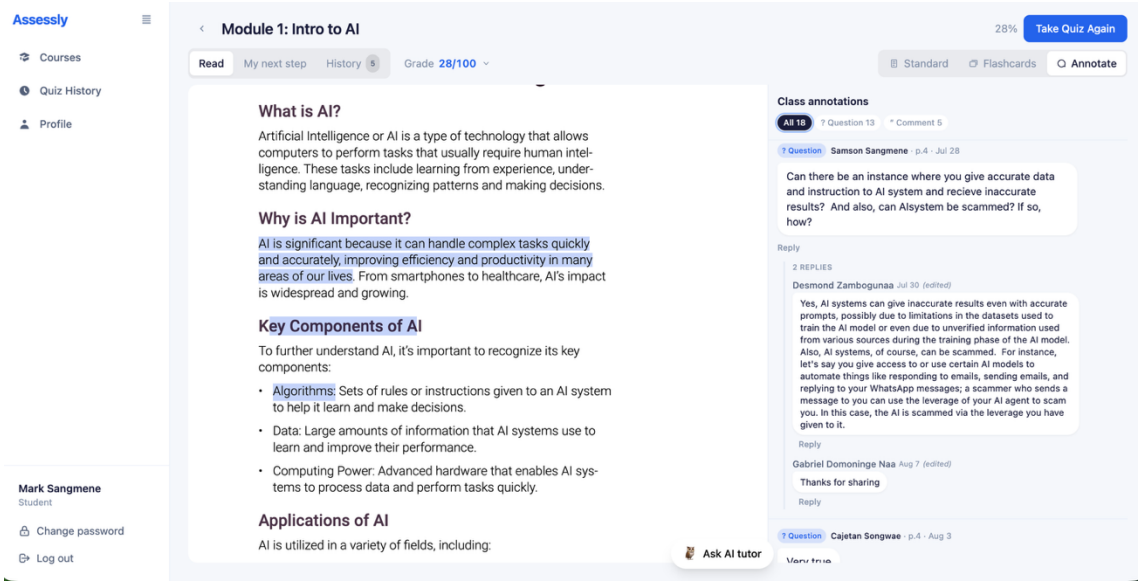


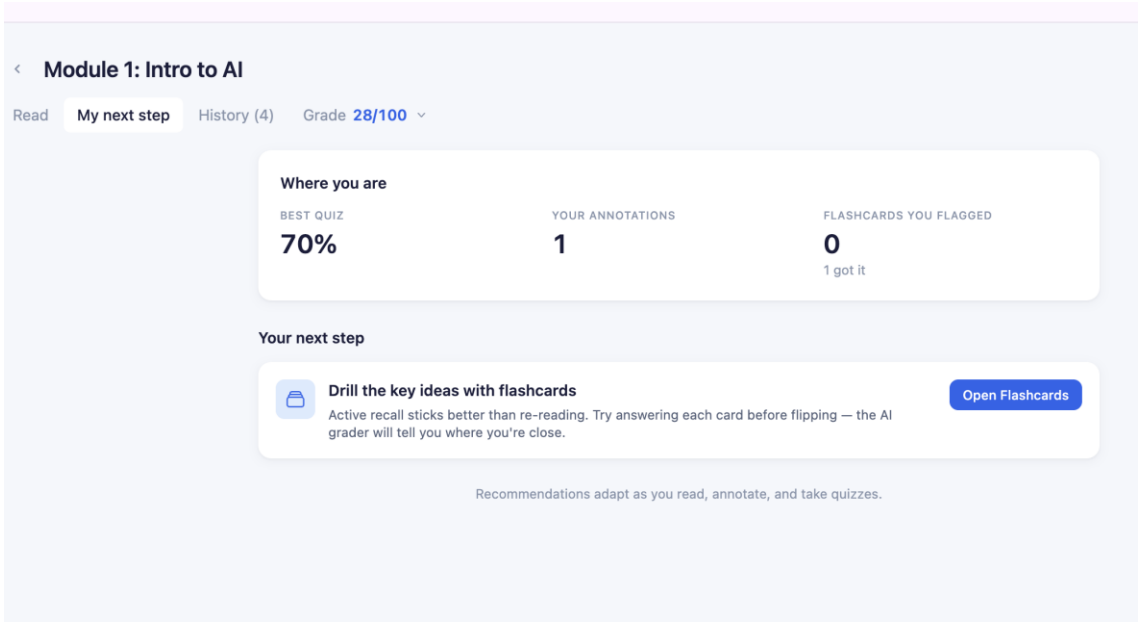




Instructor View, heatmap annotations, class pulse, etc

Student View


Assessly
Courses
Quiz History
Profile
Mark Sangmene
Student
Change password
Log out
Module 1: Intro to AI
28%
Take Quiz Again
Read
My next step
History 5
Grade 28/100
Standard
Flashcards
Annotate
What is AI?
Artificial Intelligence or AI is a type of technology that allows computers to perform tasks that usually require human intelligence. These tasks include learning from experience, understanding language, recognizing patterns and making decisions.
Why is AI Important?
AI is significant because it can handle complex tasks quickly and accurately, improving efficiency and productivity in many areas of our lives. From smartphones to healthcare, AI's impact is widespread and growing.
Key Components of AI
To further understand AI, it's important to recognize its key components:
Algorithms: Sets of rules or instructions given to an AI system to help it learn and make decisions.
Data: Large amounts of information that AI systems use to learn and improve their performance.
Computing Power: Advanced hardware that enables AI systems to process data and perform tasks quickly.
Applications of AI
AI is utilized in a variety of fields, including:
Ask AI tutor
Class annotations
All 18
Question 13
Comment 5
Question Samson Sangmene · p.4 · Jul 28
Can there be an instance where you give accurate data and instruction to AI system and recieve inaccurate results? And also, can AIsystem be scammed? If so, how?
Reply
2 REPLIES
Desmond Zambogunaa Jul 30 (edited)
Yes, AI systems can give inaccurate results even with accurate prompts, possibly due to limitations in the datasets used to train the AI model or even due to unverified information used from various sources during the training phase of the AI model. Also, AI systems, of course, can be scammed. For instance, let's say you give access to or use certain AI models to automate things like responding to emails, sending emails, and replying to your WhatsApp messages; a scammer who sends a message to you can use the leverage of your AI agent to scam you. In this case, the AI is scammed via the leverage you have given to it.
Reply
Gabriel Domoninge Naa Aug 7 (edited)
Thanks for sharing
Reply
Question Cajetan Songwae · p.4 · Aug 3
Very true


Module 1: Intro to AI
Read
My next step
History (4)
Grade 28/100
Where you are
BEST QUIZ
70%
YOUR ANNOTATIONS
1
FLASHCARDS YOU FLAGGED
0
1 got it
Your next step
Drill the key ideas with flashcards
Active recall sticks better than re-reading. Try answering each card before flipping — the AI grader will tell you where you're close.
Open Flashcards
Recommendations adapt as you read, annotate, and take quizzes.

REFERENCES

IMS Global Learning Consortium. (2019a). Learning Tools Interoperability (LTI) assignment and grade services specification (Version 2.0). https://www.imsglobal.org/spec/lti-ags/v2p0

IMS Global Learning Consortium. (2019b). Learning Tools Interoperability (LTI) core specification (Version 1.3). https://www.imsglobal.org/spec/lti/v1p3

Adesope, O. O., Trevisan, D. A., & Sundararajan, N. (2017). Rethinking the use of tests: A meta-analysis of practice testing. Review of Educational Research, 87(3), 659–701.

Agarwal, P. K., Nunes, L. D., & Blunt, J. R. (2021). Retrieval practice consistently benefits student learning: A systematic review of applied research in schools and classrooms. Educational Psychology Review, 33(4), 1409–1453.

Chi, M. T. H., Adams, J., Bogusch, E. B., Bruchok, C., Kang, S., Lancaster, M., Levy, R., Li, N., McEldoon, K. L., Stump, G. S., Wylie, R., Xu, D., & Yaghmourian, D. L. (2018). Translating the ICAP theory of cognitive engagement into practice. Cognitive Science, 42(6), 1777–1832.

Gorzycki, M., Desa, G., Howard, P. J., & Allen, D. D. (2020). “Reading is important,” but “I don’t read”: Undergraduates’ experiences with academic reading. Journal of Adolescent & Adult Literacy, 63(5), 499–508.

Kalir, R., & Garcia, A. (2021). Annotation. MIT Press.

Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., … Kasneci, G. (2023). ChatGPT for good? On opportunities and

challenges of large language models for education. Learning and Individual Differences, 103, 102274.

Ke, Z., & Ng, V. (2019). Automated essay scoring: A survey of the state of the art. Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, 6300–6308.

Kerr, M. M., & Frese, K. M. (2017). Reading to learn or learning to read? Engaging college students in course readings. College Teaching, 65(1), 28–31.

Kulik, J. A., & Fletcher, J. D. (2016). Effectiveness of intelligent tutoring systems: A meta-analytic review. Review of Educational Research, 86(1), 42–78.

Kurdi, G., Leo, J., Parsia, B., Sattler, U., & Al-Emari, S. (2020). A systematic review of automatic question generation for educational purposes. International Journal of Artificial Intelligence in Education, 30(1), 121–204.

Lewis, J. R. (2018). The System Usability Scale: Past, present, and future. International Journal of Human–Computer Interaction, 34(7), 577–590.

Miller, K., Lukoff, B., King, G., & Mazur, E. (2018). Use of a social annotation platform for pre-class reading assignments in a flipped introductory physics class. Frontiers in Education, 3, 8.

Panadero, E. (2017). A review of self-regulated learning: Six models and four directions for research. Frontiers in Psychology, 8, 422.

St Clair-Thompson, H., Graham, A., & Marsham, S. (2018). Exploring the reading practices of undergraduate students. Education Inquiry, 9(3), 284–298.

Wisniewski, B., Zierer, K., & Hattie, J. (2020). The power of feedback revisited: A meta-analysis of educational feedback research. Frontiers in Psychology, 10, 3087.

Wollny, S., Schneider, J., Di Mitri, D., Weidlich, J., Rittberger, M., & Drachsler, H. (2021). Are we there yet? A systematic literature review on chatbots in education. Frontiers in Artificial Intelligence, 4, 654924.

Zhu, X., Chen, B., Avadhanam, R. M., Shui, H., & Zhang, R. Z. (2020). Reading and connecting: Using social annotation in online classes. Information and Learning Sciences, 121(5/6), 261–271.