Bilkent University
Department of Computer Engineering

Senior Design Project

Team ID: T2521

Project Name: Ubien: The Immersive Inquiry Based Learning Platform

Project Specification Document

Group Members:

22203367, Ahmet Deniz Gelir, deniz.gelir@ug.bilkent.edu.tr
22202254, Efe Can Tatar, can.tatar@ug.bilkent.edu.tr
22201820, Kemal Onur Özkan, onurozkan@ug.bilkent.edu.tr
22001736, Eren Uslu, eren.uslu@ug.bilkent.edu.tr
22203328,Simay Uygur, simay.uygur@ug.bilkent.edu.tr

Supervisor:
Uğur Güdükbay
Course Instructors:
Atakan Erdem
Mert Bıçakçı

26.11.2025

Table of Contents

1. Introduction

It is a commonplace in the present day to hear students bemoan that while they have “learned” a great deal on paper, they have retained only a fraction of the information they gathered over the course of their education. It is natural, then, to ask: what has been the point of learning all that information if it functioned merely as a temporary exercise in memorization? The current paradigms of education, from structured lectures to solution-driven problem sets and exam-oriented milestones, are built around the efficient delivery and assessment of knowledge rather than the cultivation of genuine understanding. In such systems, students often become passive recipients, carrying information only long enough to reproduce it when asked, and quickly losing it afterward.

This widespread pattern reveals a fundamental problem: the absence of active self-discovery. Educational psychology and centuries of pedagogical tradition alike affirm that concepts uncovered through one’s own reasoning are remembered more deeply, understood more clearly, and integrated more permanently into one’s intellectual framework [1]. Yet most modern educational tools unintentionally short-circuit this process. They present polished explanations, prepackaged insights, and step-by-step solutions that resolve uncertainty too quickly, preventing learners from engaging with the material in a way that forms resilient mental models.

Ubien arises from the recognition that true understanding occurs when students do not merely receive ideas, but are made to come up with it themselves. Drawing on the principles of Inquiry-Based Learning (IBL), Ubien transforms traditional IBL textbooks into interactive courses where students iteratively advance through answering a sequence of questions. Instead of being told what to think, learners are socratically questioned into discovering why concepts hold together and how the structures of a discipline emerge organically from first principles.

In this environment, knowledge is not an external artifact handed down by an instructor, it becomes something the student builds. The result is a form of understanding that is permanent, self-sourced, and meaningfully tied to the learner’s own cognitive efforts. Ubien seeks to restore this mode of discovery-based learning to the digital age, offering a platform where students engage with material in a way that mirrors the curiosity-driven process that underlies real intellectual growth.

As we proceed through this specification, we explore how Ubien leverages technology to revive the foundational experience of learning through inquiry, providing students with a modern tool that supports not only academic success but the development of independent, rigorous, and enduring understanding.

1.1 Description

Ubien is a 28-week digital education innovation project commencing in the Fall 2025-2026 semester at Bilkent University. The project’s mission is to transform traditional inquiry-based learning materials into dynamic, self-guided online learning experiences powered by Large Language Models. The system enables students to upload IBL textbooks and automatically generate interactive, question-driven courses led by a personalized AI instructor. The platform leverages LLMs integrated with a retrieval layer to ensure factual accuracy and adaptive learning flow. To increase engagement, Ubien incorporates animated instructor embodiments capable of facial expressions, gestures, and speech synthesis: simulating the experience of learning from a real instructor. The end result is an interactive educational web platform where learners progress by answering guided questions and discovering the subject matter themselves. Ubien's key deliverables include an operational web application with backend services, a 2D/3D virtual instructor model, and a dynamic course generator that converts uploaded materials into structured inquiry sessions. Supporting deliverables, such as project specification, detailed design, and user testing reports, will be produced across seven major milestones, culminating in a final system demonstration by Week 28. By combining inquiry-based pedagogy with artificial intelligence and embodied interaction, Ubien aims to make self-directed discovery learning accessible, engaging, and cost-efficient, offering a glimpse into the future of personalized education through intelligent, interactive technology.

1.2 High Level System Architecture & Components of Proposed Solution

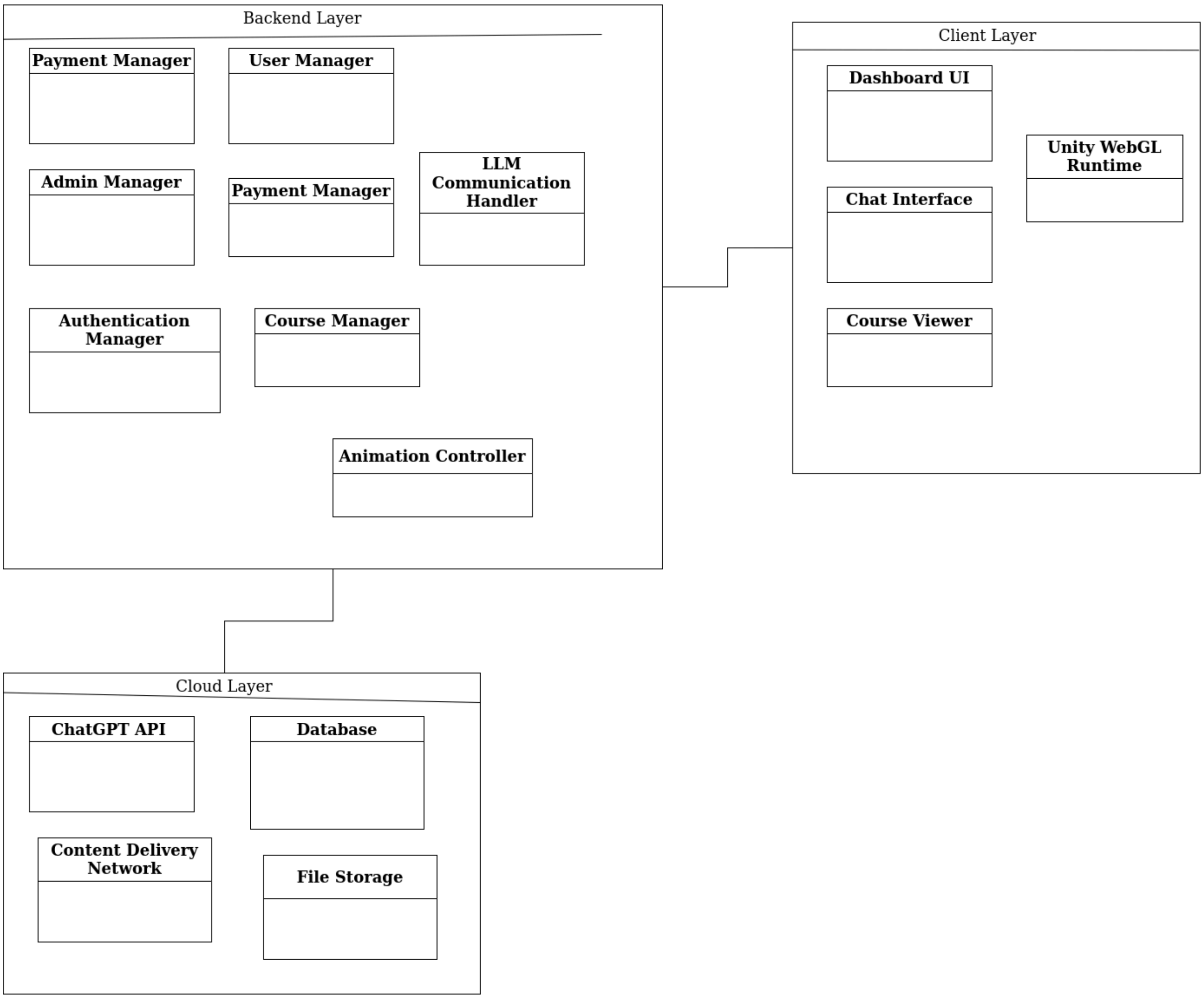

Figure 1: High-Level System Architecture

The Client Layer, Backend Layer, and Cloud Layer make up the system's three-layer architecture. Together, these layers manage user contact, LLM-powered guidance, PDF submission, course creation, and animated teacher responses, as well as chatting with the course.

1.2.1 Client Layer

The client layer delivers the system's user interface. It utilizes HTML, CSS, and JavaScript, and operates entirely within the web browser. Users can engage with the AI instructor, explore their created courses, upload textbooks, and see animated model responses.

1.2.1.1 Dashboard UI

This component displays a summary of every course the user has created. It shows access buttons, creation dates, course titles, and progress status. It serves as the user's primary point of navigation.

1.2.1.2 Chat Interface

A natural language text field is provided for communicating with the virtual instructor through the Chat Interface. Users can participate in guided IBL discourse, ask questions about the course material, respond to the instructor's instructions, and request clarifications or repetitions as needed.

1.2.1.3 Course Viewer

Displays the generated IBL course content to the user, including question prompts, slides, progress indicators, and instructor interactions. It provides the main interface through which learners navigate the course, submit answers, and track their advancement.

1.2.1.4 Unity WebGL Runtime

The Unity WebGL instructor avatar is loaded and rendered by this component. It displays lip-syncing with TTS audio, eye contact, gestures, facial expressions, and upper-body movements.

1.2.2 Backend Layer

The system's primary business logic is managed by the backend layer. It generates all computational output provided to the client, manages user accounts, stores course data, processes uploaded PDFs, and interacts with external AI services.

1.2.2.1 Authentication Manager

All user authentication and security processes, including user registration, login/logout, email verification, and credential validation, are managed by this component. It guarantees that system functions are only accessible to verified users.

1.2.2.2 User Manager

User-related data, including profile details, preferences, and progress tracking, is managed by the User Manager. It controls how users engage with stored system data and their courses.

1.2.2.3 Admin Manager

For developers or maintainers, this module offers administrative features. Log viewing, system activity monitoring, error troubleshooting, and maintenance activities are some examples of what it could involve.

1.2.2.4 Course Manager

Generated course sections, IBL-style questions, and progress checkpoints are all stored and retrieved by the course manager. Gives users and the frontend clear access to all course data.

1.2.2.5 LLM Communication Handler

This component converts outputs into IBL course content, generates conversations for the Chat Interface, delivers PDF content, prompts, and questions to the ChatGPT API, and receives organized responses, serving as the primary entry point for the system's AI capabilities.

1.2.2.6 Animation Controller

It communicates with the Unity WebGL runtime via WebSockets to synchronize lip movements with generated audio and to apply gesture and emotion metadata received from the backend, ensuring expressive and coherent instructor animations.

1.2.3 Cloud Layer

The third-party services required for AI processing, multimedia rendering, and persistent storage comprise the cloud layer. These services supply the media structure and critical processing resources that the system needs.

1.2.3.1 ChatGPT API

All of the system's AI-powered functions are carried out by this component, including extracting and analyzing content from uploaded PDFs, constructing IBL-style courses, and generating explanations and guided inquiry stages. Additionally, it generates text-to-speech audio from instructor responses and facilitates natural-language conversation during study sessions.

1.2.3.2 Content Delivery Network

Large client-side content, including avatar models, textures, animation files, JavaScript bundles, and Unity assets, is efficiently stored and served by the CDN.

1.2.3.3 File Storage

File Storage is responsible for storing all user-uploaded materials. It manages the persistent storage of IBL textbooks (PDF files) and any auxiliary resources required by the platform.

1.2.3.4 Database

Database stores all persistent system-related information, including user accounts, course metadata, generated course materials, progress monitoring, token/cost history, chat history, and system logs.

1.3 Constraints

1.3.1. Implementation Constraints

- All course generation and TTS must use the OpenAI API through the backend.

- The server must be implemented in Go, with WebSockets for live interaction.
- User accounts, courses, and progress must be stored in PostgreSQL.
- User login must rely on secure server-side sessions (cookie-based), also required for WebSocket upgrades.
- The system must run in Docker containers, both for development and deployment.
- The project must use GitHub for code, issues, and basic CI.

1.3.2. Economic Constraints

- API costs and token restrictions may limit the number of interactions that a user can make in a month of subscription.
- The API costs will be covered from the funds obtained from monthly subscriptions. Users will be allowed to use as much as they pay.
- All software will be built using freely available technologies.
- For an initial setup intended to serve up to 100 users, the monthly hosting costs amount to approximately 45 $, broken down as follows: 15 $ (backend server) + 15 $ (managed PostgreSQL DB) + 5 $ (storage) + 10 $ (bandwidth) + 1 $ (domain/DNS) + 0 $ (SSL) + 0 $ (CDN, Cloudflare Free Tier). In the long run, these hosting costs will be covered by allocating a portion of the revenue obtained from monthly subscriptions, allowing Ubien to operate sustainably and profitably.

1.3.3. Ethical Constraints

- User data and their chats must stay confidential and encrypted
- Users won’t be allowed to share chat logs and courses with others as the course may include personalized or copyrighted material for the user.

- All datasets and educational resources used will comply with fair-use, privacy, and ethical data-handling principles
- All visual and instructional material must be correctly attributed and comply with fair-use principles.

1.4 Professional and Ethical Issues

Ubien raises several professional and ethical issues due to its use of LLMs, persistent user data, and copyrighted educational materials. First, data privacy and confidentiality are critical: the system stores user accounts, progress, chat history, and uploaded textbooks. In line with the stated ethical constraints, this data must remain confidential and be protected with secure sessions, encryption, and strict access control; users should not be able to share personalized courses or chat logs that may contain private or copyrighted content.

Second, the platform must respect intellectual property and fair-use principles. Uploaded IBL books and generated materials may be subject to copyright; therefore, external resources must only be used within permitted educational contexts, and visual or instructional content must be properly attributed and not redistributed as full substitutes for the original works.

Finally, reliance on an AI tutor introduces issues of bias, reliability, and academic integrity. LLM outputs may contain mistakes or latent biases, so the system should encourage students to treat responses as guidance rather than unquestionable truth and allow instructors or developers to review and improve problematic content. The interaction design should support inquiry-based learning, not shortcutting assignments, and the development team has an ongoing professional duty to monitor system behavior, log failures, and correct defects that could harm users' learning experience.

1.5 Standards

- The UML 2.5.1 standard (ISO/IEC 19505) will be followed in all diagrams, to ensure uniform, formal, and easily recognizable documentation [2].
- The IEEE 830 Software Requirements Specification (SRS) standard is followed when writing both functional and non-functional requirements to guarantee that they are unambiguous, traceable, measurable, and precise [3].
- Management of information security principles defined in ISO/IEC 27001 will be used for user authentication data, course data, and uploaded textbook materials [4]. This covers risk-aware system behavior, restricted access, and safe storage procedures.
- To maintain professional and consistent project reports, the IEEE referencing style will be used for all academic citations and references [5].
- To ensure the proper processing of model outputs, a compliant request structure, and the responsible use of LLM capabilities, all AI-powered interactions (LLM-based course creation, PDF reading, and TTS) must adhere to OpenAI's API usage regulations [6].

- For ethical conduct in the system's design, development, and delivery, all developers will adhere to the ACM Code of Ethics and Professional Conduct [7]. This involves accountability for user privacy, openness, equity, and acknowledgment of the project's educational goal.
- WebGL will be used to run the animated teacher, adhering to W3C standards for browser-based graphics for rendering effectiveness and web browser compatibility [8].
- The system will be implemented in accordance with Clean Code standards to ensure readability, maintainability, and continuous extensibility [9].

2. Design Requirements

2.1. Functional Requirements

2.1.1 User Account Management

- The system should enable users to create an account by entering the required information, including an email address, password, age, and school level.
- The system should send a confirmation email to the user whenever a new account is created. This email should contain a verification link or code that the user must use to activate their account.
- The system should only allow users to log in after their email address has been verified. Unverified accounts shall be prevented from logging in and accessing any authenticated functionality.

2.1.2 Course Creation From IBL Books

- The system should allow users to upload an Inquiry-Based Learning (IBL) textbook in PDF format.
- The system should take the text from the uploaded PDF and process it.
- The system should compute the approximate token count needed to process the extracted PDF content.
- The system should calculate the estimated cost of creating the course based on token requirements and current API pricing.
- The program should show the computed price to the user for that course.
- The system should ask the user to confirm before creating the course.
- When the user gives their approval, the system should make a course using the ChatGPT API based on the PDF content.
- The system should be able to store all the course content that has been generated, including all sections, questions, and metadata.

2.1.3 Course Management and Access

- The system should display a list of all courses generated by the user on the dashboard.
- The system should allow users to access any of their created courses.

- The program should track the user's progress through each course, recording completed tasks and questions.
- The system should record all calculated estimates of token counts and costs related to course creation.

2.1.4 Interaction and AI Tutoring

- The system should provide a text interface in natural language, allowing users to interact with course content by asking questions, responding to the instructor, and requesting clarifications or other course-related assistance.
- The system should use the ChatGPT API to generate answers based on the course content.

2.1.5 Study Modes

- The system should offer the user the following delivery modes for selection:
  Mode A: Text Only
  Mode B: Body Animation
  Mode C: Body + Facial Animation
- When Mode A is selected, the system displays instructor responses as on-screen text only.
- When Mode B is chosen, the system should generate an upper-body avatar in a neutral pose and with neutral gestures.
- When Mode C is chosen, the system should generate an avatar with facial expressions, gaze, and emotional animations that mirror the instructor's character.
- The system should synchronize the avatar's mouth movement with the TTS audio for Modes B and C.

2.1.6 Speech and Animation Control

- The system should use a text-to-speech service to generate audio for the instructor's dialogue in animation modes.
- The system should produce animation-related data (covering timing, expressions, and gestures) based on the LLM output.
- The system should display animations and play audio in real time in the browser.

2.1.7 Error Handling

- The system should detect invalid or unreadable PDF files during upload and show an error message to the user. The system must prevent the creation of a course if the uploaded PDF is invalid.
- The system should alert the user if the PDF exceeds the maximum token limit permitted for processing.
- The system should alert the user if ChatGPT or TTS API failures occur during processing.

2.2. Non-Functional Requirements

2.2.1. Usability

- The system should enable an average user, without a technical background, to upload a PDF, initiate a course, and interact with the study interface within two minutes of initial use.
- All core actions, for instance, uploading textbooks, accepting the cost, selecting study modes, and asking questions, should be accessible in no more than four clicks from the dashboard.

2.2.2. Reliability

- In the event of network failures, API timeouts, or invalid uploads, the system displays an error message within ten seconds and allows users to resubmit the request with descriptive error messages.
- The backend system will maintain an availability of at least 99% during regular operational hours.
- User progress, including completed tasks, questions asked, and lessons viewed, will be saved instantly and remain secure, even if the browser is refreshed or closed.

2.2.3. Performance

- Under typical network conditions, only text ChatGPT API responses will be delivered within three to five seconds.
- Once the user approves the cost, the system will initiate the course generation process within one minute.
- Text extraction and token estimation for PDFs up to 50 MB should take 10 seconds or less on the backend.
- The system will perform text summarization, content chunking, and filtering to avoid exceeding the ChatGPT API token limit or context buffer size.
- WebGL-based animations will maintain a minimum of 25 frames per second on mid-range laptops with 8 GB of RAM and an integrated GPU.

2.2.4. Supportability

- Each of the system's distinct modules, PDF parsing, token estimation, LLM processing, and animation control, can be upgraded or changed without impacting TTS, and animate the system as a whole.
- The system should record failed API requests, PDF extraction errors, and token estimation failures, along with the timestamps and request details. To facilitate maintenance and troubleshooting, logs must be retained for a minimum of 30 days.
- API keys, price values, and animation choices should all be customizable using environment variables rather than being hard-coded. This will eliminate the need for recompilation of the backend.

2.2.5. Scalability

- The backend should be able to run multiple instances simultaneously and support 200 active users without any loss of performance.
- To reduce response time by at least 50% for repeated requests, frequently reused LLM outputs, such as textbook summaries and chapter metadata, will be cached.
- The system should load the list of courses within two seconds when a user opens the dashboard, even if the database contains a large number of courses, with a total of 10,000 or more courses stored.

3. Feasibility Discussions

3.1. Market & Competitive Analysis

There are many other AI based learning platforms in the market that aim to make education more accessible, cheaper and easier. As Ubien, we aim to separate ourselves from the competition by focusing on personalization of the learning experience and by adapting an inquiry based approach to education. The most similar product in the market right now is an AI teaching tool named YouLearn. This service allows the users to upload pdf files, slides or video links and select specific parts of them. Then the user can ask for the tool to create flashcards, quizzes, summaries or take notes about this[10]. YouLearn also has the functionality to turn on voice mode which reads out the text and listens in to your voice for commands. Although there is no instructor model to make the voice mode more interactive, this is the closest on-market product to what we are creating. The main separator between our project and YouLearn will be the inquiry based approach that our LLMs will take and the interactive nature of the instructor model. Furthermore YouLearn prices its service at 20$ per month for a single account which makes the barrier for entry very steep. This service shows that it is possible to make a learning product that allows for personalized teaching and voice recognition, however Ubien's main features will focus on a different kind of learning strategy so the products will not be copies of each other.

There are some other competitors that are very similar to YouLearn that promise the same exact features like flashcard and summary creations. Namely Turbo AI and Mindgrasp are direct competitors to YouLearn and therefore our project as well. These both host the exact same functionalities that YouLearn has and therefore are in the same competitive department to our project's goals. One important remark about these products is Turbo AI shares its user count with the public which shows that it has reached 5 million total users [11]. This is largely due to Turbo AI's free trial phase that makes entry very easy. TurboAI also has lower and regional pricing, starting at 150TL monthly for Turkish users. Mindgrasp also has lower pricing starting at 10$ monthly with a 3 day free trial. These competitors show that the same service can be provided for way cheaper prices and that there is a big market that searches for effective learning tools that can help them in their studies or interests.

Another interesting competitor is called Eduaide. This service is instead designed for teachers to create learning materials for their students. Although this is not a direct competition to our product, it has many different functionalities that might become parts of our product in the future as well. For teaching, it has curriculum alignment and structured lesson planning features that automatically give a rough estimate of how a class might go and how it can be taught in good order. This functionality in itself is a proof of concept for our product since the IBL approach will also create a learning roadmap and important points list in the background and then guide the user through the course by itself as well. Eduaide also has many other functionalities such as creating educational games and organizing notes, but these functionalities are not in line with the scope of our project. Priced at 6$ for the premium version with a free version as well, Eduaide also shows that teaching tools can be brought to life for cheaper prices. So although Eduaide is not a direct competitor to our product, it stands as a great proof of concept as to how LLMs can be used to create learning material.

In conclusion there are many other AI-assisted teaching tools in the market, but none of them exactly have the same functionalities as Ubien does. The number of competitors and the features they host shows the feasibility of the project while the similar nature of all the competitors shows the uniqueness of Ubien as a product in the AI assisted learning space.

3.2. Academic Analysis

The research we conducted on the viability of AI-assisted teaching platforms showed that these services can in fact enhance the learning experience of users of all levels [12]. Therefore this makes Ubien relevant and noteworthy enough to develop as a service to feed the needs of learners around the world.
The viability of Ubien was also researched from academic sources for feasibility reasons. The main points of interest in this feasibility research is whether or not the main functionalities can be developed and put into use in a relatively smooth framework. This includes the integration of an AI-guided digital teacher model, which has been proven to be viable and useful in regards to the learning experience and the correctness of the knowledge served with the help of systems to neutralize AI hallucinations before in peer papers [13]. The effectiveness of using AI for IBL methods has also been tested before with positive results for the students, although the research was only conducted on language learning skills, which will only make up a small part of Ubien’s capabilities [14]. Lastly, the interactive teacher model with human like animations and speech has been proven possible by many in-use applications and papers before. Mainly, it was our innovation expert’s paper that showed that it was possible to integrate human-like models into LLM-based education agents and have positive results on engagement and perception of the lesson given [15].

These findings and previous studies point at Ubien being a viable product in today's market with the increasing interest and need for AI-assisted learning tools. Our application can be utilized by the general public to learn about any topic they might be interested in a way that is engaging and easy to access.

5. Glossary

LLM: Large Language Model
TTS: Text-to-Speech

CI: Continuous integration
CDN: Content Delivery Network

6. References

[1] I. Kaiser, J. Mayer, and D. Malai, “Self-Generation in the Context of Inquiry-Based Learning,” *Frontiers in Psychology*, vol. 9, 2018. doi: 10.3389/fpsyg.2018.02440.

[2] “About the Unified Modeling Language Specification Version 2.5.1,” *www.omg.org*. https://www.omg.org/spec/UML/2.5.1/

[3] “IEEE SA - IEEE 830-1998,” *IEEE Standards Association*. https://standards.ieee.org/ieee/830/1222/ (accessed Nov. 20, 2025).

[4] ISO, “ISO/IEC 27001,” *ISO*, 2022. https://www.iso.org/standard/82875.html (accessed Nov. 20, 2025).

[5] “IEEE Editorial Style Manual,” *IEEE Author Center Journals*. https://journals.ieeeauthorcenter.ieee.org/create-your-ieee-journal-article/create-the-text-of-your-article/ieee-editorial-style-manual/ (accessed Nov. 20, 2025).

[6] OpenAI, “Usage policies,” *Openai.com*, 2024. https://openai.com/policies/usage-policies/ (accessed Nov. 20, 2025).

[7] Association for Computing Machinery, “ACM Code of Ethics and Professional Conduct,” *Association for Computing Machinery*, Jun. 22, 2018. https://www.acm.org/code-of-ethics (accessed Nov. 20, 2025).

[8] W3C, “Standards - W3C,” *W3.org*, 2019. https://www.w3.org/standards/ (accessed Nov. 20, 2025).

[9] R. C. Martin, *Clean Code a Handbook of Agile Software craftmanship.* Prentice Hall, 2008.

[10] YouLearn Inc., “YouLearn AI,” YouLearn, https://www.youlearn.ai/ (accessed Nov. 28, 2025).

[11] [1] Turbo AI, “We Just Hit 5 Million Users: Here's Our Story,” https://www.turbo.ai/blog/october-2025-announcement-5-million-users (accessed Nov. 28, 2025).

[12] A. Renato, I. Nadia, and Francisco, “Challenges and Opportunities of AI-Assisted Learning: A Systematic Literature Review on the Impact of ChatGPT Usage in Higher Education,” *International Journal of Learning, Teaching and Educational Research*, vol. 22, no. 7, pp. 122–135, Jul. 2023, doi: https://doi.org/10.26803/ijlter.22.7.7.

[13] Z. Zhao, Z. Yin, J. Sun, and P. Hui, “Embodied AI-Guided Interactive Digital Teachers for Education,” pp. 1–8, Nov. 2024, doi: https://doi.org/10.1145/3680533.3697070.

[14] H.-C. Yeh, “The synergy of generative AI and inquiry-based learning: transforming the landscape of English teaching and learning,” *Interactive learning environments*, pp. 1–15, Apr. 2024, doi: https://doi.org/10.1080/10494820.2024.2335491.

[15] S. Sonlu, B. Bendiksen, F. Durupinar, and U. Güdükbay, “The Effects of Embodiment and Personality Expression on Learning in LLM-based Educational Agents,” *arXiv.org*, 2024. https://arxiv.org/abs/2407.10993