INFORMATION SYSTEMS EDUCATION JOURNAL

Volume 25, No. 1
January 2027
ISSN: 1545-679X

In this issue:

4. **CurriculumGPT: Supporting Curriculum Design with AI-Augmented Tooling**
Kareem Dana, West Texas A&M University
Jeffry Babb, West Texas A&M University

23. **A Comparative Case Study of AI-Enabled and Traditional Experiential Simulations in Operations and Supply Chain Education**
Yan Jin, Quinnipiac University
Ae-Sook Kim, Quinnipiac University
Tan Gürpinar, Quinnipiac University
Iddrisu Awudu, Quinnipiac University

37. **Personality, Organizational Environment, and the Use of Generative AI: A Conceptual Framework for Understanding Human-AI Interaction**
David Firth, University of Montana
Simon Laub, Business Academy Aarhus
Elizabeth Kohl, University of Montana
Fang Chen, University of Montana

55. **Adapting CAMIL for K–12 World Language Learning: A Qualitative Study of Immersive VR in Middle School Spanish**
Emily Loughlin, Nichols College
Kevin Mentzer, Nichols College

66. **Developing an Artificial Intelligence Course for a Small Undergraduate Program**
Scott Turner, University of Richmond
Lionel Mew, University of Richmond
Terry Turner, University of Richmond

iscap
Information Systems & Computing Academic Professionals

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **ISCAP** (Information Systems and Computing Academic Professionals). Publishing frequency is five times per year. The first year of publication was 2003.

ISEDJ is published online (https://isedj.org). Our sister publication, the Proceedings of the ISCAP Conference (https://iscap.us/proceedings) features all papers, abstracts, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the ISCAP conference. All papers, whether award-winners or not, are invited to resubmit for journal consideration after applying feedback from the Conference presentation. Award winning papers are assured of a publication slot; however, all re-submitted papers including award winners are subjected to a second round of three blind peer reviews to improve quality and make final accept/reject decisions. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is under 35%.

Information Systems Education Journal is pleased to be listed in the Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org. Special thanks to volunteer members of ISCAP who perform the editorial and review processes for ISEDJ.

2026 ISCAP Board of Directors

Amy Connolly James Madison University President	Michael Smith Georgia Institute of Technology Vice President	Jeff Cummings Univ of NC Wilmington Past President
David Firth University of Montana Director	Mark Frydenberg Bentley University Director/Secretary	Leigh Mutchler James Madison University Director
RJ Podeschi Millikin University Director/Treasurer	Bryan Reinicke Rochester Institute of Technology / Director	Jeffry Babb West Texas A&M University Director/Curricular Matters
Eric Breimer Siena University Director/2026 Conf Chair	Thomas Janicki Univ of NC Wilmington Director/Meeting Planner	Xihui "Paul" Zhang University of North Alabama Director/JISE Editor

Copyright © 2026 by Information Systems and Computing Academic Professionals (ISCAP). Permission to make digital or hard copies of all or part of this journal for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial use. All copies must bear this notice and full citation. Permission from the Editor is required to post to servers, redistribute to lists, or utilize in a for-profit or commercial use. Permission requests should be sent to Kevin Mentzer, Editor, editor@isedj.org.

INFORMATION SYSTEMS EDUCATION JOURNAL

Editors

Kevin Mentzer
Editor
Nichols College

Ira Goldman
Associate Editor
Siena University

David Yates
Associate Editor
Bentley University

Michelle Louch
Teaching Cases & Exercises
Editor
University of Pittsburgh - Greensburg

Mark Pisano
Teaching Cases & Exercises
Associate Editor
Southern Connecticut
State University

Thomas Janicki
Publisher
Univ of NC Wilmington

David Woods
Assistant Publisher
Miami University
Regionals

Paul Witman
Emeritus Editor
(2021-2026)
California Lutheran
University

Jeffry Babb
Emeritus Editor
(2016-2021)
West Texas A&M
University

Donald Colton
Emeritus Editor
(2003-2010)
Brigham Young University
Hawaii

CurriculumGPT: Supporting Curriculum Design with AI-Augmented Tooling

Kareem Dana
kdana@wtamu.edu

Jeffry Babb
jbabb@wtamu.edu

West Texas A&M University
Canyon, TX 79016, USA

Abstract

A key challenge in model curriculum development, especially for information systems, is its lengthy revision cycle, often spanning 5 to 10 years, and static reports that quickly become outdated. Recognizing this, recent model curricula, CC2020 and IS2020, call for a more sustained and continuous development process. In response, we present CurriculumGPT, a prototype AI tool built with a retrieval-augmented generation (RAG) architecture, to help realize that goal and address limitations of a prior tooling effort, the Computing Competencies Curricula Tool. We conducted a pilot study to evaluate the system across representative curriculum design tasks, including fact retrieval, comparative analysis, and content creation. Our training corpus included syllabi, course materials, learning objectives, and model curricula reports. Responses were assessed with a standardized human-applied rubric and the automated RAG evaluation framework, RAGAS, which evaluated quantitative metrics of answer faithfulness, relevancy, context precision, and recall. Results indicate that CurriculumGPT performed reliably, with improved performance achieved through metadata-aware chunking, a refinement to standard RAG approaches that preserves structural relationships in the data. Our findings show how AI and RAG systems can be adapted to support curriculum design while highlighting limitations and challenges that establish a foundation for future research in this emerging area.

Keywords: Curriculum Design, Generative AI, Computing Education, Competency, Curriculum Tooling, AI in Education

Recommended Citation: Dana, K., Babb, J., (2026). CurriculumGPT: Supporting Curriculum Design with AI-Augmented Tooling. *Information Systems Education Journal*, v25(n1) pp 4-22. DOI# https://doi.org/10.62273/CVEC6056

CurriculumGPT: Supporting Curriculum Design with AI-Augmented Tooling

Kareem Dana and Jeffry Babb

1. INTRODUCTION

Computing education continually evolves, shaped by emerging technologies and industry demands. As the field advances, so does the debate over what graduates should know and be able to do. As a result, developing model curricula in computing and information systems is both essential and inherently challenging.

Recent initiatives to guide this process include the Computing Curricula 2020 (CC2020) and IS2020 reports (Clear et al., 2020; Leidig et al., 2020). These reports strongly advocate for a move towards a competency-based computing curriculum which emphasizes what graduates should be able to do rather than just what they know (Babb et al., 2025). Both reports, however, are ultimately static documents produced through lengthy development cycles.

Recognizing this limitation, both CC2020 and IS2020 call for a more continuous, community-driven approach supported by effective software tooling. IS2020 envisions a living document community where educators collaboratively update curricula over time (Babb et al., 2021; Clear et al., 2020; Leidig et al., 2020).

Initial efforts to realize this vision included the Computing Competencies Curricula Tool or C3T, developed through a collaboration between the Information Systems & Computing Academic Professionals (ISCAP) and the Association for Computing Machinery (ACM). C3T demonstrated the feasibility of a web-based platform for creating and comparing competencies. However, it relied heavily on manual data entry, offered limited interactivity, and contained only a small initial dataset. These limitations constrained its ability to support scalable and continuous curriculum development (C3T, 2025).

This highlights a broader problem. There is currently no effective tooling that supports dynamic, data-driven curriculum design by leveraging the large volume of unstructured curricular artifacts already available to educators, such as syllabi, course catalogs, and learning objectives. Recent advances in generative AI, particularly large language models (LLMs) and retrieval-augmented generation (RAG) offer a promising foundation to address this challenge.

Following a design science approach, we developed and evaluated CurriculumGPT, an AI-augmented curriculum design tool built on a RAG architecture tailored to the structure of curricular data. In contrast to generic RAG implementations, our approach incorporates metadata-aware chunking to preserve relationships within curricular data, improving retrieval fidelity. The system supports natural language interaction enabling educators to query, analyze, and generate curriculum content more efficiently, while also providing source attribution to support transparency and explainability of responses.

This paper makes three contributions:

- First, we present CurriculumGPT, our AI tool, which applies a RAG architecture to support curriculum development tasks using academic data.
- Second, we introduce and evaluate metadata-aware chunking as a refinement for RAG systems operating on hierarchical curricular data, demonstrating its impact on retrieval accuracy and task performance.
- Third, we provide initial design insights into the use of AI-powered tools for curriculum design, including considerations for transparency, explainability, and human oversight, highlighting both their potential and limitations through an exploratory pilot study, and discuss future research paths in this area.

Together, these contributions extend prior work on curriculum tooling by demonstrating how AI can be adapted to the specific challenges of curriculum design, while also contributing to emerging design knowledge on the application of RAG in structured, domain-specific contexts.

2. LITERATURE REVIEW

IS2020 (Leidig et al., 2020) defines a competency as "the graduate's ability to apply knowledge, skills, and dispositions to effectively complete

tasks." In this definition, knowledge is the most familiar component, referring to factual elements embedded in syllabi and course materials that students should know. Skills represent the ability to apply that knowledge in practice, often aligned with Bloom's taxonomy to capture cognitive complexity (e.g., remember, analyze, create). Dispositions encompass the soft skills such as ethics, professionalism, and values that guide how knowledge and skills are applied in context (Babb et al., 2025; Leidig et al., 2020). To support software tooling, Babb et al. (2020) introduced the eXtensible Computing Curriculum Reporting Language (XCCRL), a structured data format for representing competencies and their components. Building on this foundation, C3T was developed as a web-based platform to help educators build competencies (C3T, 2025).

Much of the recent literature on AI in education has focused on enhancing the student learning experience. Marquardson (2024) explored how AI tools can support self-directed learning, while Mew (2024) examined the pedagogical role of ChatGPT in higher education. Lang and Gürpinar (2025) built a RAG chatbot to enhance student learning in an online course. Dickey & Bejarano (2024) developed GAIDE, a generative AI-powered assistant designed to help instructors draft course content and generate teaching materials.

Other studies have begun exploring how generative AI can support curriculum development itself. Padovano & Cardamone (2024) proposed a structured approach for using AI to define engineering competencies, while Sridhar et al. (2023) conducted an empirical study on the effectiveness of large language models (LLMs) in generating course-level learning objectives. Iweuno et al. (2024) discuss using AI to improve curriculum design by enabling data-driven decision making, personalized learning, and more inclusive curricular content. Lastly, Chai et al. (2025) performed a thorough review of other generative AI uses in instructional design. These studies suggest that AI can play a meaningful role in instruction and curriculum design. However, to our knowledge no previous work has explicitly developed and empirically evaluated AI tools with a focus on IS curricula.

Finally, Eduglopedia, though now defunct, represented an early attempt to create a global, open encyclopedia of educational programs and curricular resources through a collaborative web-based platform. It built an impressive collection of content that led to the release of annual reports, which shared trends, insights, and guidance with the IS community. This effort demonstrated the need and interest in software support for curriculum design (vom Brocke et al., 2020).

3. CASE STUDY: THE C3T TOOL

Purpose and Vision of C3T

One of the key ideas emphasized in the CC2020 and IS2020 reports is that curriculum development should be a sustained and ongoing process, not a one-time event every few years (Clear et al., 2020; Leidig et al., 2020). These reports, along with the work of Babb et al. (2021), call for moving away from traditional, static curriculum models in favor of more flexible and dynamic models. C3T was developed as an initial step towards realizing this vision. Its primary purpose was to offer educators and other stakeholders a web-based platform through which they can build, visualize, and compare competencies (C3T, 2025).

A secondary, yet more ambitious goal was to serve as a community-supported platform for the collective stewardship of model curricula (C3T, 2025). With such a community, C3T could realize the vision of a continuous model curriculum that adapts quickly to emerging innovations. For example, the topic of artificial intelligence is not highlighted explicitly as a competency in the IS2020 report. With a living document community in place, educators could have already collaboratively proposed, reviewed, and integrated AI-related competencies. These contributions could then be easily shared across universities well before publication of the next model curriculum.

Architectural Overview

C3T was developed as a software-as-a-service application to support the creation, management, and comparison of curriculum competencies. Development occurred incrementally between 2019 and 2024 through the part-time work of IS faculty and student developers (C3T, 2025).

Early development focused on basic competency management, including create, read, update, and delete functionality. Subsequent enhancements added search, filtering, and table-based comparison features (C3T, 2025).

A later development phase envisioned C3T as a community-driven platform incorporating collaboration, role-based governance, and discussion forums. Preliminary work was completed on user registration. However, many of the critical community features remain

incomplete (C3T, 2025).

Outcome and Lessons Learned

In November 2024, C3T was publicly deployed and presented at an academic conference. Development and deployment yielded several important insights. First, the tool succeeded in demonstrating the feasibility of a web-based platform for modeling computing competencies. However, since its deployment, user engagement has been minimal. While the website received visitors, there is no evidence of significant user registration, collaboration, or competency creation (C3T, 2025).

Several factors appear to contribute to this lack of adoption. First, the process of entering a single competency involves multiple steps and decisions, with significant manual data entry. This is a high effort process that likely discourages visitors. Second, C3T contains little existing data. Like a new social media platform, it lacks the critical mass of content needed to make participation worthwhile. Third, the user interface is minimal and lacks competency visualizations or interactive elements to help users interpret, compare, and make decisions about curricula. Finally, the living document features have not been implemented, limiting C3T's potential to support such a community. These limitations underscore our motivation to explore AI-augmented curriculum tooling as the next stage of tooling development.

4. CURRICULUMGPT

Drawing on the limitations of C3T and the characteristics of curricular data, we developed CurriculumGPT with four key design goals:

1. CurriculumGPT should intelligently analyze documents such as syllabi, course catalogs, and learning objectives to automatically extract relevant content such as competencies, reducing the need for manual data entry.
2. Users should be able to interact with CurriculumGPT using conversational questions such as *"Which competencies align with this course?"* or *"What's missing from our cybersecurity curriculum?"*. Such interactions can increase engagement and enable more informed decision making.
3. The tool should preserve the structure within curricular data. Such data contains important relationships across courses, learning objectives, and competencies, which must be maintained during retrieval.
4. Users should be able to trace responses back to source materials to assess accuracy and build trust.

Architecture Overview

We developed a prototype implementation of CurriculumGPT based on these design goals. Our architecture is grounded in design science research (Hevner et al., 2004), which emphasizes building artifacts to address identified problems and evaluating their effectiveness, and information retrieval principles, particularly the RAG paradigm (Lewis et al., 2020).

Our prototype is built using Streamlit, LlamaIndex, and OpenAI's GPT-4o LLM. Streamlit is a Python framework that provides easy-to-use web interfaces for users and easily integrates with LLMs. See Appendix A for a screenshot of CurriculumGPT that illustrates Streamlit and the user interface.

LlamaIndex is the RAG pipeline used in our prototype. RAG allows the system to generate responses based on curated academic documents, rather than relying solely on the LLM's general pretraining (Lewis et al., 2020). LlamaIndex searches the curated documents and passes the most relevant chunks to the LLM. This addresses a critical challenge in using general-purpose LLMs for education tooling: lack of contextual awareness about specific curricula or local academic data (Lang & Gürpinar, 2025). Our prototype includes explainability features by citing the specific documents retrieved from the training data during each query. Finally, the LLM used is GPT-4o. See Figure 1 below for a diagram of the CurriculumGPT architecture.

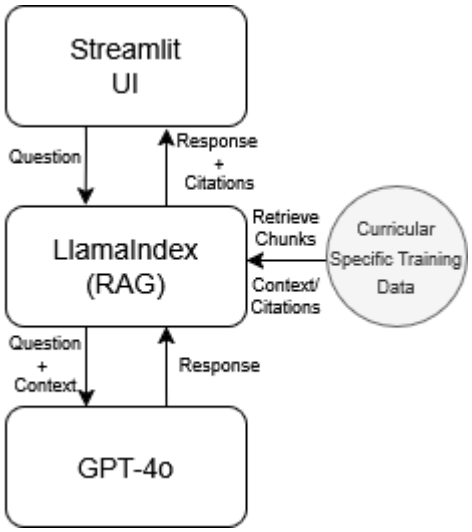


Figure 1: CurriculumGPT Architecture

Type of Data	Number of Documents (43 documents)	Used in CurriculumGPT	Purpose for Inclusion
Model Curricula	2 Documents (136,461 words)	IS2020, CC2020	Grounds responses in standards; ensure alignment with models.
Course Materials	36 documents (129,103 words)	Course catalogs, degree plans, syllabi, Course Learning Objectives (CLOs)	Supports queries about specific courses, course comparisons, and demonstrates how courses are structured.
Accreditation standards	1 document (6,614 words)	ABET criteria	Supports accreditation tasks; helps faculty align content with accreditation requirements.
Competency Frameworks	3 documents (4,988 words)	Matrices mapping skills, knowledge elements, dispositions to competencies	Structures competency relationships; enables comparisons and gap analysis.
Taxonomies	1 document (384 words)	Bloom's Taxonomy	Guides phrasing of objectives at appropriate cognitive levels.

Table 1: Training Data Used in CurriculumGPT

While RAG has been widely applied in many domains, our architecture adapted and refined it for curriculum design. Curricular data is inherently hierarchical and relational, with dependencies across courses, learning objectives, and competencies that are not well suited to naive chunking or retrieval. To preserve those relationships and improve retrieval accuracy, we implemented metadata-aware chunking, tagging each chunk with its associated course name or relationship (Danter et al., 2024; Qu et al., 2024). The RAG architecture also supported the implementation of source tracing and updates to the training corpus without requiring model retraining.

Implementation Details

To support reproducibility and provide greater transparency of our system, key technical aspects are described below.

The training corpus consisted of curricular materials from a single academic department, including course syllabi, course catalogs, degree plans, and course learning objectives. It also included model curriculum reports (IS2020 and CC2020), as well as competency frameworks and taxonomies (e.g., Bloom's Taxonomy). In total, the corpus included forty-three documents. Refer to Table 1 for the complete training data details.

Documents were segmented using a sentence-based chunking strategy. Each document was divided into chunks of approximately 1024 tokens, with an overlap of 200 tokens between adjacent chunks. A token is a word or part of a word used by language models during processing and generation. To improve retrieval accuracy, a metadata-aware chunking mechanism was implemented. Course and hierarchy identifiers were embedded directly into each chunk, ensuring semantic continuity across chunks.

Chunk retrieval was performed using a vector similarity search with the top 10 chunks selected per query. Additionally, a curriculum domain-specific re-ranking step was applied as part of the metadata-aware chunking technique to prioritize chunks whose metadata matched the requested curricular component (e.g., course or learning objective). This hybrid retrieval approach helped improve precision.

Responses were generated using the OpenAI GPT-4o model. A default temperature setting of 0.5 was used. The model was prompted with both retrieved documents and a master prompt guiding it to prioritize relevant, source-based responses while leveraging its general LLM knowledge when necessary.

The source code for CurriculumGPT is available at: https://github.com/kareemy/CurriculumGPT/

5. EVALUATION AND RESULTS

To evaluate CurriculumGPT, we designed a series of queries across three categories of tasks: basic retrieval, comparative analysis, and content creation. Queries were developed to represent realistic tasks faced by curriculum designers, based on examples drawn from faculty

Task Category	Faithfulness w/o and with		Response Relevancy		Context Recall		Context Precision	
Retrieval	0.79	0.96	0.92	0.96	0.73	0.97	0.64	0.90
Comparison	0.46	0.82	0.87	0.89	0.61	0.83	0.45	0.71
Creation	0.76	0.79	0.94	0.94	N/A		0.61	0.78

Table 2: Average RAGAS Evaluation Metrics without and with Metadata-aware Chunking

Task Category	Faithfulness (Accuracy)	Relevance	Clarity	Actionability
Retrieval	4.50	4.67	4.58	4.67
Comparison	3.50	3.83	4.00	4.00
Creation	3.83	4.50	3.75	3.50

Table 3: Average Reviewer Rubric Scores (1 to 5)

experiences and model curriculum guidelines. Twenty queries and responses from each category (60 total) were evaluated using RAGAS, a standardized, automated framework for evaluating RAG+LLM systems across four quantitative metrics: faithfulness, response relevancy, context recall, and context precision (each scored 0-1).

Faithfulness measures how factually consistent a response is with the training data. A score of 1 indicates that the response is fully supported by the training data, while lower scores suggest hallucinations or that the tool is answering based on its general LLM knowledge instead of the curriculum-specific corpus. Response relevancy measures how directly and appropriately the response addresses the query. RAGAS measures this by reverse-engineering the question from the response and comparing it to the original, with lower scores indicating incomplete or off-topic answers. (Es et al., 2024)

Context recall measures how well CurriculumGPT retrieves and incorporates relevant information from the training data. RAGAS performs this evaluation by comparing the response to a correct, reference response. A low score suggests that the response is missing important information. Finally, context precision measures the proportion of retrieved chunks that were relevant. Low precision means many irrelevant or unused chunks were retrieved. These metrics are established measures, with full methodological details documented on the RAGAS website and supporting publications (Es et al., 2024).

The results with and without metadata-aware chunking are summarized in Table 2. Lower scores (left) were without metadata-aware chunking. In addition to automated metrics, four independent reviewers evaluated the responses using a standardized rubric (Appendix B). The rubric scored responses on a 1-5 Likert scale for faithfulness, relevance, clarity, and actionability. Appendix C includes queries and responses that are representative of each category.

Reviewers were faculty in our IS department and selected based on their familiarity with curriculum design. Reviewers were given guidance on how to interpret each dimension. To assess the consistency of reviewer evaluations, we calculated inter-rater reliability using the intraclass correlation coefficient, ICC(2,k), for each evaluation dimension. Results indicated good agreement for faithfulness (ICC = 0.88), relevance (ICC = 0.84), and actionability (ICC = 0.82), and moderate agreement for clarity (ICC = 0.74), suggesting that the rubric was applied consistently across reviewers.

System Validation: Retrieval and Fact Recall
In this first category, CurriculumGPT was prompted to retrieve specific facts from the training data. It performed reliably. It was able to retrieve accurate course learning objectives and descriptions from syllabi. It was also able to correctly define important curricular terms such as "knowledge-skill pair" from the appropriate source material. As shown in Table 2, this category achieved the highest overall scores across the RAGAS metrics, averaging above 0.9 with metadata-aware chunking. Reviewers also rated these responses highly on the rubric.

Comparisons: Syllabus Gap Analysis and Competency Comparison
This category involved more complex tasks, including asking CurriculumGPT to identify competency coverage gaps across courses and compare content. CurriculumGPT scored lower across both the RAGAS metrics and reviewer evaluations compared to the other categories. These tasks were most prone to hallucinations, but also most improved by metadata-aware

chunking. Future technical tweaks and a larger training corpus could improve these types of responses even further.

Creation: Authoring Competencies and Course Outlines

In tasks asking CurriculumGPT to create new content, it performed relatively well. The faithfulness metric was naturally lower since the tool generated content not explicitly present in the training data and context recall was not measured because these tasks did not have a definitive reference answer. In this case, the response relevancy metric and reviewer scores were the best indicators of usefulness. The system produced plausible draft competencies and outlines for new courses (high response relevancy), demonstrating its utility as a brainstorming aid or starting point for curriculum planning. However, reviewers indicated its output often created broad competencies containing multiple learning objectives that were more complex than necessary. Improved prompting and follow-up queries can help refine these results. For example, follow-up queries such as "*Separate this competency into three distinct competencies.*" or "*Revise using Bloom's taxonomy at the analyze level.*" led to improved results.

Benefits & Faculty Use Cases

One of the most immediate use cases of CurriculumGPT is its ability to create course-level competencies and learning objectives. Faculty can use its responses as a starting point, but refinement will probably be needed to produce clear, appropriately scoped competencies. These results are consistent with the research of Sridhar et al. (2023) who also studied whether LLMs can generate learning objectives.

We also found that our prototype may be beneficial in helping faculty explore ideas for new courses or revise existing ones. Leveraging its training data, faculty can ask questions such as, "*Generate a sample syllabus outline for a course that teaches cybersecurity and maps to the CC2020 competencies."* or *"I'm revising our course on Data Analytics. Based on current industry trends and existing competencies in our program, what new topics or learning objectives should I consider adding?"* This result is consistent with recent research on the topic of using LLMs for curriculum design (Iweuno et al., 2024; Padovano & Cardamone, 2024).

Overall, these results demonstrate that CurriculumGPT shows promise as a curriculum support tool. However, it has limitations, challenges remain, and future work is needed.

6. LIMITATIONS AND FUTURE WORK

Limitations

This study represents an initial, exploratory evaluation of CurriculumGPT and contains several limitations. First, the system was evaluated using curricular data from a single department within one university. This narrow scope limits the generalizability of the findings and does not fully capture how the system would perform across different universities, disciplines, or curriculum structures. While the results provide a proof of concept and highlight promising architectural design choices, future work should examine system performance using larger and more diverse datasets.

Second, the evaluation relies, in part, on RAGAS metrics. These metrics evaluate text generation quality, not curriculum design or educational quality. As a result, high RAGAS scores should not be interpreted as evidence of improved curriculum design outcomes. Instead, they indicate that responses are well-grounded in retrieved source material and aligned with the input query. This distinction is important, as effective curriculum design requires domain expertise and holistic judgment beyond what automated metrics can capture.

Third, four human reviewers were included to complement the RAGAS evaluation. While inter-rater reliability analysis indicated moderate to good agreement, a larger and more diverse group of reviewers would provide a more robust assessment of the system.

Finally, while we compared standard RAG with metadata-aware chunking, we did not include a baseline using an unguided LLM without RAG, nor comparisons to existing curriculum design tools or human-only workflows. As a result, the findings should be interpreted as evidence of relative improvement within the RAG-based approach rather than a comprehensive evaluation of alternative methods. Future research could incorporate more comparisons to better assess the effectiveness of AI-augmented curriculum design tools.

Future Work

The results of this pilot study reveal several opportunities to improve CurriculumGPT, particularly in supporting more complex curriculum tasks such as comparison and gap analysis.

Knowledge graphs provide a way to keep the training data interconnected and maintain relationships between courses, learning objectives, and competencies. Incorporating knowledge graphs to explicitly model curricular relationships could enable the system to better respond to those more complex queries (Ji et al., 2021).

Additionally, integrating table-augmented generation (TAG) or text-to-SQL capabilities could further improve the tool by allowing it to interface directly with structured databases (Biswal et al., 2024). Together, these extensions point to the need for a mechanism to coordinate how structured and unstructured curricular data can be accessed within the system.

Another promising direction is the Model Context Protocol (MCP), an open standard that allows LLMs to interact with external data systems (Karimova & Dadashova, 2025). Adopting MCP could allow CurriculumGPT to interface with both knowledge graphs and SQL databases through a single, standardized protocol. This architecture could also lay the groundwork for agentic workflows, where the tool can autonomously query live data or execute complex analysis tasks. Similar MCP-based approaches have already been demonstrated in the structural engineering domain (Avila et al., 2025).

Improving explainability remains an important area for future work. We implemented source tracing and inline citations within CurriculumGPT, but more explainability is needed. Exploring toolkits like TruLens (TruLens, n.d.) could be valuable. TruLens integrates within the RAG pipeline to validate retrieval accuracy and offer interactive explanations to help users trace how the system arrived at a response.

We also see several other applications for CurriculumGPT or AI curriculum tools that can be explored in future work. One is its use as a support tool in accreditation-related tasks, such as ABET or AACSB reviews. When trained on accreditation criteria and an institution's courses, CurriculumGPT could help map course content to accreditation criteria or synthesize data needed for a report. Another potential application is training the tool on job postings to extract desired skills and trends in industry, which could inform and align model curricula with workforce expectations.

Adding these capabilities would be an important step toward realizing the living document community vision as outlined in IS2020. While CurriculumGPT is a first step, that vision remains aspirational and will depend on this future work.

7. AI RISKS AND GOVERNANCE

This section focuses on the broader implications of a tool like CurriculumGPT, highlighting risks and governance practices we believe are necessary to ensure responsible and successful use of AI in curriculum design.

Using an LLM in educational settings introduces several ethical and institutional risks that must be considered carefully. At the heart of that discussion is the training data. An LLM's ability to generate accurate and domain-specific responses depends on the quality of the training data (Lang & Gürpinar, 2025; Lin et al., 2024).

Decisions about what training data to include can greatly impact the quality of the system's output and introduce systemic bias or blind spots if not carefully reviewed (Ladwig & Schwieger, 2024; Lin et al., 2024). For example, an LLM could suggest outdated terminology, reinforce gender stereotypes, or fail to account for regional legal and cultural differences in topics like cybersecurity or ethics.

There are also privacy concerns associated with indexing internal university documents, particularly if those documents contain sensitive information. Likewise, LLMs may be pre-trained on copyrighted material, raising legal and ethical questions about reusing that content in applications like CurriculumGPT (Ladwig & Schwieger, 2024).

Another major concern is AI hallucination, the generation of plausible but inaccurate information, which could mislead users relying on the tool for curriculum decisions (Ladwig & Schwieger, 2024; Lin et al., 2024).

Future development of AI curriculum tools should proceed with a clear governance structure that focuses on ethical design and human oversight. AI tools to improve productivity are here to stay and they will find their way into curriculum design, either by us or others. Only with human oversight can these risks be properly managed.

Prior work, including IS2020 (Leidig et al., 2020) and Babb et al. (2021), outlines potential governance structures that include a board of directors and various task forces. We support these models and suggest they add a dedicated AI oversight representative or task force to ensure that AI is used responsibly and trained on

a diverse set of data that accurately reflects the will of the IS community.

8. CONCLUSIONS

This paper presents CurriculumGPT as an initial step towards supporting curriculum design with AI-augmented tooling. Our evaluation showed that the tool performed reliably on retrieval tasks, achieving high scores across all four RAGAS metrics (above 0.9) and strong reviewer ratings (above 4.5/5.0). Comparative tasks, such as identifying competency gaps across courses, were more challenging, but accuracy improved notably with metadata-aware chunking. For content creation, the system generated plausible draft competencies and course outlines that reviewers found useful for brainstorming, though results were noted as being too broad at times.

The introduction of metadata-aware chunking and source attribution are two key technical contributions of this work. They addressed a limitation of standard RAG approaches and illustrated how RAG architectures can be adapted to better support domain-specific tasks such as curriculum design.

Perhaps more importantly, this work establishes a foundation for future research. Exploring new, richer ways to represent curricular data and more advanced approaches to retrieval can further enhance the system's usefulness for IS educators.

Building tools like CurriculumGPT and further research contribute towards the living document vision articulated in IS2020. Realizing this vision will require continued collaboration among educators, researchers, and professional organizations to ensure that these tools are designed, used, and governed in ways that reflect the needs and values of the IS community.

9. REFERENCES

Avila, C., Ilbay, D., & Rivera, D. (2025). Human-AI Teaming in Structural Analysis: A Model Context Protocol Approach for Explainable and Accurate Generative AI. *Buildings*, 15(17), 3190. https://doi.org/10.3390/buildings15173190

Babb, J., Gardner, L., Leidig, P. M., Salmela, H., Sooriamurthi, R., Muthupoltotage, U. P., & Qutab, S. (2021). Towards a Continuous Process for Using and Sustaining IS Curriculum Guidelines. *Proceedings of the 2021 AIS SIGED International Conference on Information Systems Education and Research*. 3. https://aisel.aisnet.org/siged2021/3

Babb, J., Sharp, J. H., Waguespack, L., Abdullat, A., & Dana, K. (2020). eXtensible Computing Curriculum Reporting Language (XCCRL). *Information Systems Education Journal*, 18(6), 13-27.

Babb, J., Yates, D., & Waguespack, L. (2025). On Becoming: Why Disposition Distinguishes Information Systems Education from Training. A Commentary on Model Curricula. *Information Systems Education Journal*, 23(1). https://doi.org/10.62273/JRSO9158

Biswal, A., Patel, L., Jha, S., Kamsetty, A., Liu, S., Gonzalez, J. E., Zaharia, M. (2024). Text2sql is not enough: Unifying ai and databases with tag. arXiv preprint arXiv:2408.14717.

C3T. (n.d.). C3T: Computing Competencies Curricula Tool. Retrieved June 22, 2025, from https://c3t.dev/

Chai, D. S., Kim, H. S., Kim, K. N., Ha, Y., Shin, S. S. H., & Yoon, S. W. (2025). Generative Artificial Intelligence in Instructional System Design. *Human Resource Development Review*. https://doi.org/10.1177/15344843251320256

Clear, A., Parrish, A., Ciancarini, P., Cuadros-Vargas, E., Frezza, S., Gal-Ezer, J., Impagliazzo, J., Pears, A., Takada, S., Topi, H., van der Veer, G., Vichare, A., Waguespack, L., Wang, P., Zhang, M. (2020). Computing Curricula 2020 (CC2020), *Paradigms for Future Computing Curricula*, ACM, IEEE, https://doi.org/10.1145/3467967

Danter, D., Mühle, H., & Stöckl, A. (2024). Advanced Chunking and Search Methods for Improved Retrieval-Augmented Generation (RAG) System Performance in E-Learning. *Human Factors in Design, Engineering, and Computing*, *159*(159). https://doi.org/10.54941/ahfe1005756

Dickey, E., & Bejarano, A. (2024, October). GAIDE: A Framework for Using Generative AI to Assist in Course Content Development. *In 2024 IEEE Frontiers in Education Conference* (FIE) (pp. 1-9). IEEE. https://doi.org/10.1109/fie61694.2024.10893132

Es, S., James, J., Anke, L. E., & Schockaert, S. (2024, March). RAGAS: Automated evaluation of retrieval augmented generation. *In Proceedings of the 18th Conference of the European Chapter of the*

Association for Computational Linguistics: System Demonstrations (pp. 150-158). https://doi.org/10.18653/v1/2024.eacl-demo.16

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv*:2312.10997, 2(1).

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, 75-105. https://doi.org/10.2307/25148625

Iweuno, B. N., Orekha, P., Ojediran, O., Imohimi, E., & Adu-Twum, H. T. (2024). Leveraging Artificial Intelligence for an inclusive and diversified curriculum. *World Journal of Advanced Research and Reviews*, 23(2), 1579-1590. https://doi.org/10.30574/wjarr.2024.23.2.2440

Ji, S., Pan, S., Cambria, E., Marttinen, P., & Philip, S. Y. (2021). A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE transactions on neural networks and learning systems*, *33*(2), 494-514. https://doi.org/10.1109/tnnls.2021.3070843

Karimova, S., & Dadashova, U. (2025). The Model Context Protocol: A Standardization Analysis for Application Integration. *UNEC Journal of Computer Science and Digital Technologies*, 1(1), 50-59. https://doi.org/10.30546/UNECCSDT.2025.01.050

Ladwig, C., Schwieger, D. (2024). An Eye Toward the Softer Side of CC2020 Computing Curricula: Professional, Legal, and Ethical Artificial Intelligence Issues. *Information Systems Education Journal*, 22(1), pp.53-88. https://doi.org/10.62273/FFDJ1170

Lang, G., Gürpinar, T. (2025). AI-Powered Learning Support: A Study of Retrieval-Augmented Generation (RAG) Chatbot Effectiveness in an Online Course, *Information Systems Education Journal* v23 (n2) pp 4-13. https://doi.org/10.62273/ZKLK5988

Leidig, P., Salmela, H, & The Joint ACM/AIS IS2020 Task Force. (2020). IS2020: A Competency Model for Undergraduate Programs in Information Systems, ACM, AIS, https://doi.org/10.1145/3460863

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33, 9459-9474.

Lin, Z., Guan, S., Zhang, W., Zhang, H., Li, Y., & Zhang, H. (2024). Towards trustworthy LLMs: a review on debiasing and dehallucinating in large language models. *Artificial Intelligence Review*, 57(9), 243. https://doi.org/10.1007/s10462-024-10896-y

Marquardson, J., (2024). Embracing Artificial Intelligence to Improve Self Directed Learning: A Cybersecurity Classroom Study. *Information Systems Education Journal*, 22(1), pp.4-13. https://doi.org/10.62273/WZBY3952

Mew, L., Money, W.H., (2024). Leveraging ChatGPT for Higher Education Course Offerings. *Information Systems Education Journal* 22(5) pp 72-80. https://doi.org/10.62273/OKLK1331

Padovano, A., & Cardamone, M. (2024). Towards human-AI collaboration in the competency-based curriculum development process: The case of industrial engineering and management education. *Computers and Education: Artificial Intelligence*, 7, 100256. https://doi.org/10.1016/j.caeai.2024.100256

Qu, R., Tu, R., & Bao, F. (2024). Is Semantic Chunking Worth the Computational Cost? In *Findings of the Association for Computational Linguistics*: NAACL 2025, pp 2155–2177, Albuquerque, NM. Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.findings-naacl.114

Sridhar, P., Doyle, A., Agarwal, A., Bogart, C., Savelka, J., & Sakr, M. (2023). Harnessing llms in curricular design: Using gpt-4 to support authoring of learning objectives. *arXiv preprint arXiv:2306.17459*.

TruLens. (n.d.). TruLens: Evaluating and tracking LLM apps. Retrieved June 19, 2025, from https://www.trulens.org/

vom Brocke, J., Tan, B., Topi, H., Weinmann, M. (2020), AIS Global IS Education Report, The Global Report of the Association for Information Systems on Information Systems Education 2020. 4th Edition. Available at: https://www.researchgate.net/publication/339738830_AIS_Global_IS_Education_Report_201920

APPENDIX A

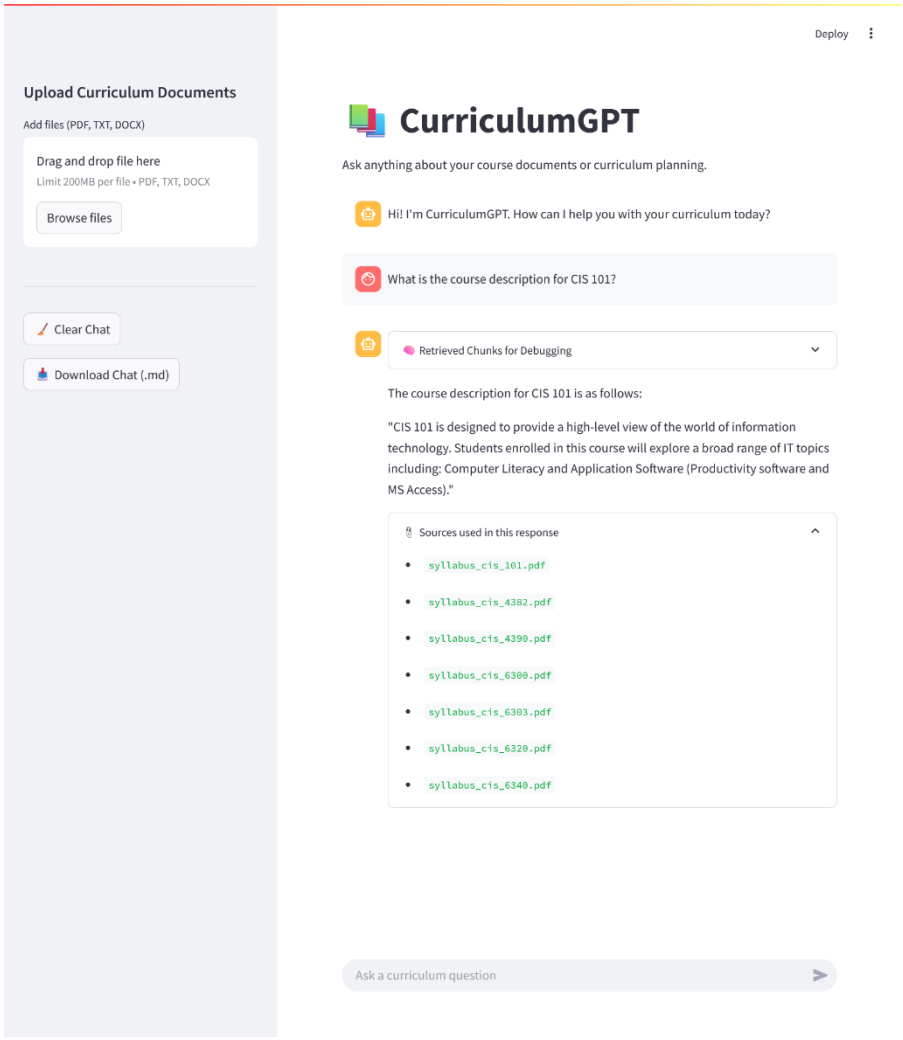


Figure 2: Screenshot of CurriculumGPT

This screenshot shows the CurriculumGPT user interface. Users can interact with it through this web interface by typing questions, seeing responses and citations. Users can also use the sidebar to upload documents, clear, or download chats.

APPENDIX B

CurriculumGPT Evaluation Rubric

1. Faithfulness (Accuracy)
   Does the response stay true to the source documents or knowledge base, avoiding hallucinations or invented details?
   1 = Incorrect: The response contains significant inaccuracies or information that contradicts the source documents.
   2 = Mostly Incorrect: The response contains multiple factual errors, making it unreliable.
   3 = Partially Correct: The response is a mix of correct and incorrect information; some key facts are missing or wrong.
   4 = Mostly Correct: The response is generally accurate but may contain minor inconsistencies or omissions.
   5 = Completely Correct: The response is fully accurate and aligns with the provided source documents.

2. Relevance
   Does the response directly address the query and stay on topic?
   1 = Irrelevant: The response does not address the query and provides completely unrelated information.
   2 = Mostly off-topic: The response touches on the topic but fails to provide a direct or useful answer to the query.
   3 = Somewhat relevant: The response contains some relevant information but also includes extraneous details or misses key aspects of the query.
   4 = Mostly relevant: The response is on topic and provides a useful answer, but could be more focused.
   5 = Fully relevant to the query: The response is directly on topic and provides exactly the information needed to answer the query.

3. Clarity
   Is the response well-written, logically structured, and easy to understand?
   1 = Very unclear/confusing: The response is difficult to read and understand.
   2 = Hard to follow: The response lacks a clear logical flow and may be too generic or off-topic to be truly helpful.
   3 = Somewhat clear, needs work: The response is generally understandable but may be too broad.
   4 = Clear with good structure: The response is well-written, easy to follow, and appropriately scoped.
   5 = Very clear, polished, and coherent: The response is exceptionally well-written and perfectly tailored to the request. It is concise yet comprehensive.

4. Actionability
   Would this response be practically helpful for IS educators (e.g., in creating competencies, CLOs, or curriculum analysis)?
   1 = Not Actionable: The response is vague, generic, or so poor that it cannot be used for any practical curriculum task.
   2 = Minimally Actionable: The response provides very little value and would require extensive human effort to become useful.
   3 = Partially Actionable: The response provides a starting point, but significant editing, refinement, or additional research would be needed to make it useful.
   4 = Mostly Actionable: The response is a good draft that could be used with minor revisions to complete a curriculum task.
   5 = Highly Actionable: The response is a finished or near-finished product that can be used immediately to complete a curriculum task with no or minimal changes.

5. Please comment on any strengths of the AI response and weaknesses or issues: _____

APPENDIX C

Selected CurriculumGPT Queries and Responses

Category 1: System Validation - Basic Retrieval and Fact Recall

Query 1:

"What are the course learning objectives (CLOs) for CIS 3312?"

Response 1:

The Course Learning Objectives (CLOs) for CIS 3312, as outlined in the syllabus, are as follows:

1. **Create data-bound controls for web user interfaces** - This objective focuses on the ability to develop user interfaces that are dynamically linked to data sources.

2. **Separate presentation and business logic in web applications** - This objective emphasizes the importance of distinguishing the different roles in web application architecture, particularly in maintaining cleaner code and enhancing maintainability.

3. **Evaluate business logic with unit tests** - This objective aims to ensure that students are able to apply testing methodologies to the business logic within web applications to verify that the code behaves as expected.

These CLOs align with the overarching goals of critical thinking and functional business analyses, ensuring that students not only learn technical skills but also apply them effectively within the context of web application development (Source: Syllabus for CIS 3312).

Query 2:

"What is a knowledge-skill pair?"

Response 2:

A knowledge-skill pair (K-S pair) is a fundamental concept in competency-based education and curriculum design, integrating both the knowledge component and the corresponding skill level required to successfully perform a specific task or competency.

Definition and Components:

1. **Knowledge Element**: This represents the theoretical or factual information that a learner must understand or know in order to engage in a specific context or task. For example, in computing, a knowledge element could be the "principles of software development."

2. **Skill Level**: This refers to the degree of proficiency or cognitive level associated with applying that knowledge. The skill levels are often classified using Bloom's Taxonomy, which includes levels such as:

- Remember (Level 1)
- Understand (Level 2)
- Apply (Level 3)
- Analyze (Level 4)
- Evaluate (Level 5)
- Create (Level 6).

For instance, a K-S pair could be structured as follows:

- Knowledge Element: "Purpose of readable source code"
- Skill Level: "Understand" (Level 2)

Importance in Curriculum Design:

Knowledge-skill pairs are used to define competencies within educational programs, indicating what learners should know and be able to do in various situations. They enable educators to design courses and assessments that align with specific learning objectives. Each competency includes multiple K-S pairs that encapsulate the knowledge and skills necessary for mastery of that competency.

In essence, K-S pairs help clarify the relationship between what students need to learn (knowledge) and how they need to demonstrate that learning (skill). This structured approach ensures a comprehensive and coherent curriculum that effectively prepares learners for practical applications of their education.

[Reference: IS2020 Curriculum Document; CC2020 Computing Curricula Report]

Query 3:

"What is the course description for CIS 101?"

Response 3:

The course description for CIS 101 is as follows:

"CIS 101 is designed to provide a high-level view of the world of information technology. Students enrolled in this course will explore a broad range of IT topics including: Computer Literacy and Application Software (Productivity software and MS Access)."

Category 2: Comparisons - Syllabus Gap Analysis and Competency Comparison

Query 1:

"What key computing competencies are missing from the CIS 1301 syllabus?

Response 1:

To identify key computing competencies that may be missing from the CIS 1301 syllabus based on the information provided, we can look at the course description, objectives, and materials in conjunction with contemporary computing education frameworks. Here's a structured analysis:

Overview of CIS 1301 Competencies:
From the syllabus, the primary focus of CIS 1301 is on developing proficiency in Microsoft Office applications (Word, PowerPoint, Excel, and Access) and general computer literacy. The competencies related to personal productivity software aim to build foundational skills that are essential for business and academic activities.

Analysis of Missing Competencies:
1. Critical Thinking and Problem Solving:
- The syllabus emphasizes proficiency in software applications, but there is less attention on developing critical thinking skills, particularly in the context of analyzing and solving problems using IT. According to CC2020 (CC2020, page 42), 81% of employers report a lack of critical thinking and analytical skills among graduates. Integrating objectives that require students to apply software for data analysis or project management could fill this gap.

2. Collaboration and Communication Skills:
- While the course focuses on productivity software, it should also emphasize collaborative tools and skills. Competencies that encourage teamwork (e.g., using tools within Microsoft

Teams or collaborative features in Office 365) are essential in modern workplaces and can help address the communication gap noted by employers (CC2020, page 42).

3. Data Literacy:
- Given the pivotal role of data in contemporary IT environments, a syllabus that encourages understanding of data management principles, along with decisions based on data analysis, is vital. This includes exploring how to structure databases and querying with MS Access, aligning with competencies highlighted in CC2020 regarding data management (CC2020, page 42).

4. Ethics and Professional Practice:
- There is no mention of ethics in technology use, which is increasingly crucial in IT education. Integrating discussions about responsible use of information technology, data privacy, and the implications of technology in society could enhance the curriculum significantly (CC2020, page 46).

5. Emerging Technologies:
- While the course covers basic software applications, it may be beneficial to introduce competencies related to understanding and applying emerging technologies, such as cloud computing, to provide students with a more forward-thinking skill set that aligns with industry needs (CC2020, page 42).

6. Project Management Fundamentals:
- Understanding basic project management concepts, especially as they relate to IT and software projects, would be beneficial. This could be integrated through practical assignments that require planning and executing a project using the tools covered in the syllabus (CC2020, page 46).

7. Practical Application of Skills:
- There is an emphasis on proficiency in software packages, but the syllabus could benefit from more focus on the application of these tools in real-world scenarios. Assessments could include projects that require synthesizing skills across different applications to solve a problem.

Recommendations:
To align CIS 1301 with current competencies in computing, consider the following additions to the syllabus:
- Critical Thinking Projects: Include tasks that require students to synthesize information and make data-driven decisions using the software.
- Collaborative Assignments: Use tools that facilitate teamwork and enhance communication skills.
- Data Literacy Modules: Incorporate database management concepts and data analysis projects using MS Access.
- Ethical Discussions: Integrate conversations about the ethical use of technology.
- Emerging Technology Insights: Familiarize students with trends such as cloud storage and social media analytics.
- Project Management Principles: Introduce basic project management frameworks relevant to IT.

In conclusion, enhancing the CIS 1301 syllabus with these competencies would provide a more comprehensive educational experience, better preparing students for the evolving demands of the information technology landscape.

Query 2:

"Compare the competencies between CIS 4308 and CIS 4350."

Response 2:

To compare the competencies between CIS 4308 (Introduction to Data Analytics) and CIS 4350 (Data Mining Methods), we first need to examine the primary focus and competencies associated with each course.

CIS 4308: Introduction to Data Analytics

- **Competencies**:
  - Understanding foundational concepts in data analytics, including data collection, preprocessing, and exploratory data analysis.
  - Proficiency in statistical analysis and interpretation of data results.
  - Familiarity with data visualization techniques and tools to communicate findings effectively.
  - Basic skills in programming or software used for analytics (e.g., Python, R, SQL).
  - Application of analytical methods to solve business-related problems.

CIS 4350: Data Mining Methods

- **Competencies**:
  - Advanced understanding of different data mining techniques such as classification, regression, clustering, and association rule mining.
  - Proficiency in using data mining software and frameworks (like R, RapidMiner, or WEKA).
  - Ability to handle large datasets and understand underlying data structures.
  - Skills in developing predictive models and evaluating their performance.
  - Application of ethical considerations and the implications of data mining in business contexts.

Comparison of Competencies

1. **Foundational vs. Advanced**:
   - CIS 4308 focuses on foundational data analytics skills and general data interpretation, making it suitable for beginners or students looking to understand the basics of analytics.
   - CIS 4350 delves into advanced data mining techniques, suitable for students eager to specialize in predictive analytics and deeper technical skills.
2. **Technical Skills**:
   - Both courses likely require some technical skills but to different degrees. CIS 4308 may introduce students to analytics languages and tools, while CIS 4350 would require mastery of these tools alongside complex data mining algorithms.
3. **Application and Problem Solving**:
   - In CIS 4308, students will learn to analyze data to inform decisions and provide insights that can guide business strategy.
   - CIS 4350 equips students with the skills necessary to create models that predict outcomes based on historical data, ideal for roles in data science and machine learning.

4. **Ethics and Impact**:
    - While both courses will likely touch on ethical considerations related to data usage, CIS 4350 may go deeper into the implications of data mining and predictive analytics on privacy and decision-making processes.

In summary, CIS 4308 serves as an entry-level introduction to data analytics, focusing on basic skills and foundational knowledge, while CIS 4350 builds on that foundation by exploring more complex methodologies in data mining, requiring a stronger technical background and analytical capabilities. Each course targets different stages in a student's education and career path within the field of data analytics and information systems.

Category 3: Authoring Competencies and Course Outlines

Query 1:

"Can you create one focused computing competency for my course CIS 3312 that targets a single learning objective?"

Response 1:

Certainly! Here is a focused computing competency for CIS 3312 – Advanced Business Programming:

Computing Competency for CIS 3312 - Advanced Business Programming

Competency Statement:

Students will be able to apply object-oriented programming principles to develop software solutions that address defined business requirements.

Learning Objective:

Apply object-oriented principles (e.g., classes, inheritance, and encapsulation) to design and implement a functional software component that meets specified business requirements.

Query 2:

"Generate a sample syllabus outline for a course that teaches cybersecurity and maps to the CC2020 competencies."

Response 2:

Here is a sample syllabus outline for a course titled "Introduction to Cybersecurity," structured to align with the competencies defined in the CC2020 report. This syllabus is intended to cover the fundamental principles of cybersecurity, emphasizing both technical skills and the necessary communication and strategic thinking required in the field.

Course Title: Introduction to Cybersecurity

Course Description:

This course provides an introduction to the fundamental concepts, principles, and practices of cybersecurity. It covers risk management, the application of security principles, threat identification, and the skills necessary to develop and enforce security policies in various organizational contexts.

Course Objectives:

Upon completion of this course, students will be able to:

1. Evaluate the purpose and function of cybersecurity technologies and their role in organizational risk management.

2. Implement tools and systems to minimize cybersecurity threats.
3. Develop and articulate the policies and procedures necessary for responding to cybersecurity incidents.
4. Analyze risks and recommend mitigation strategies based on best practices in cybersecurity.

Week-by-Week Outline:

Week 1: Introduction to Cybersecurity
- Overview of cybersecurity
- Importance of cybersecurity in today's environment (CSEC2017, IS2020)
- Review of major cyber threats and vulnerabilities

Week 2: Cybersecurity Principles
- Fundamental cybersecurity principles: confidentiality, integrity, and availability
- Understanding risks and their impact on organizations (ITE-CSP-A)

Week 3: Cybersecurity Technologies
- Overview of cybersecurity tools and technologies (e.g., firewalls, intrusion detection systems)
- Evaluation of technologies that reduce risk of data breaches (ITE-CSP-A)

Week 4: Implementing Security Measures
- Practical implementation of security measures
- Hands-on lab: Setting up a firewall and intrusion detection system (ITE-CSP-B)

Week 5: Risk Management Strategies
- Introduction to risk management concepts
- Developing a risk management plan (ITE-CSP-C)

Week 6: Incident Response and Recovery
- Understanding incident response frameworks
- Developing response plans tailored towards high-value systems (ITE-CSP-D)

Week 7: Developing Security Policies
- Crafting and communicating effective security policies
- Case study analysis of organizational failures due to inadequate policies (ITE-GPP-C)

Week 8: Legal and Ethical Considerations in Cybersecurity
- Overview of cybersecurity laws and regulations
- Ethical implications of cybersecurity practices (ITE-GPP-C, IS2020 Competency 8)

Week 9: Social Engineering and Human Factors
- Understanding the role of human behavior in cybersecurity
- Techniques for mitigating human-related vulnerabilities

Week 10: Securing Data and Information
- Best practices for data security: protection of data at rest and in transit (ABET Criteria)
- Hands-on lab: Designing a secure database (ITE-IMA-B)

Week 11: Emerging Threats in Cybersecurity
- Analysis of current and emerging threats
- Discussion of security trends and future challenges in cybersecurity

Week 12: Final Projects and Presentations
- Students present their individual or group projects evaluating a cybersecurity framework of their choice
- Focus on lessons learned and future improvements (CSEC2017, IS2020)

Assessment:
- Participation and attendance
- Weekly quizzes
- Midterm exam
- Final project presentation
- Written report on the final project

Credits: 3 credits
Prerequisites: Introduction to Information Technology or equivalent knowledge.

References:
1. ACM. (2017). Cybersecurity Curricula 2017.
2. CC2020. (2020). Computing Curricula 2020.
3. IS2020. (2020). Information Systems Curriculum Guidelines.

This syllabus outline is structured to provide students with the foundational competencies necessary for success in the cybersecurity field while integrating practical applications and case studies for a comprehensive learning experience. The competencies related to risk management, legal considerations, and technical implementations are emphasized, aligning with the goals outlined in the CC2020 report.