

Redesigning publisher slide decks with ChatGPT: A design science approach to instructional artifact development

Steven A. Schilhabel, *University of Wisconsin Oshkosh, schilhabels@uwosh.edu*

Abstract

This paper introduces a design science methodology for creating a reusable workflow that utilizes generative AI software ChatGPT and Python programming to enhance textbook PowerPoint presentations. Traditional publisher slide decks often contain dense text and complete sentences, resulting in a poor visual hierarchy and ineffective instructional delivery. This solution merges prompt engineering techniques with automation scripts to create concise slides with strong visual appeal that adhere to modern design standards. The artifact created in a senior IS capstone course enables educators to update slides quickly, thereby decreasing students' cognitive load and aligning with best presentation practices. AI-enhanced tools enable scalable advancements in instructional design through this workflow demonstration.

Keywords: design science research; generative AI; ChatGPT; instructional design; slide deck redesign

Introduction

Modern-day instructors must create engaging and structured course content while navigating tight scheduling demands, larger class sizes, and administrative responsibilities. Textbook publishers provide prebuilt slide decks for instructional use but typically do not meet current presentation design standards. Presenters often find slides filled with text written in full sentences, making them more suitable for reading than showing them during presentations. Educators often spend considerable time revising and adjusting these decks for multiple courses every semester.

This paper addresses a common, time-intensive challenge: This work proposes a method to convert large-scale publisher-provided slide decks into streamlined teaching materials that adhere to best practices in presentation design. Our design science solution utilizes generative AI and lightweight scripting to automate the majority of the work. The modular workflow artifact combines prompt engineering with a Python script, enabling instructors to quickly transform slide decks into presentation-ready versions that meet their specific needs with minimal manual work.

The primary objective of this project is to enhance instructional slide quality while freeing faculty members from the time-consuming task of slide revision. This process is especially useful to educators who use standardized textbooks or must deliver the same course content in different sections. Implementing this solution in an IS capstone course reveals artificial intelligence as a viable design partner for advancing educational instruction.

Theoretical Foundations and Related Work

Instructional slide deck design requires expertise in several areas, such as cognitive psychology, instructional design principles, and educational technology applications. Research increasingly shows that instructional materials must minimize cognitive load while following visual design principles to maintain clarity and engage audiences.

Sweller's Cognitive Load Theory (1988) focuses on managing learners' working memory during instruction delivery. Publisher-provided slides often present verbose content alongside visual clutter, which hinders students from processing and retaining essential concepts. Mayer's 2009 study on Multimedia Learning reinforces these principles by demonstrating that the coherence principle eliminates unnecessary material, while the signaling principle emphasizes important points, and the segmenting principle divides content into manageable sections to enhance visual communication.

Experts in slide design, including Reynolds (2012) and Duarte (2008), support presentation methods that utilize minimal text, powerful visual elements, and storytelling presentations. Research in instructional communication confirms that learners remember more when instructors encourage storytelling and interaction through slides, rather than reading them word-for-word.

Design science research creates a framework for constructing and assessing artifacts that address practical challenges within educational and organizational environments (Hevner et al., 2004; Gregor & Hevner, 2013). It focuses on ensuring artifacts remain relevant through iterative refinement while maintaining their practical utility in specific contexts. In this project, an AI-enhanced slide redesign workflow is guided by DSR principles: The workflow design responds to identified instructional inefficiencies while building on established pedagogical theory and undergoes testing within authentic classroom settings.

Developing advanced generative AI systems, such as large language models, ChatGPT has created new possibilities for automating instructional tasks (Mollick & Mollick, 2023). AI usage in education is expanding rapidly but established methods for systematically applying these tools to enhance teaching materials, such as slides, are limited.

The research paper bridges this deficiency by combining AI technologies and prompt engineering alongside Python scripting to create a complete instructional workflow. Together, these theoretical foundations provide a rationale for the workflow's design and the pedagogical goals it supports: The workflow decreases educators' workload while improving student understanding and ensuring instructional materials follow evidence-based design standards.

Research Method: Design Science Approach

The study employs Design Science Research (DSR) methodology to create and evaluate an instructional artifact for redesigning textbook slide decks using generative AI. DSR represents a prominent research methodology in information systems and educational fields, focusing on developing innovative artifacts to address real-world issues through rigorous evaluation (Hevner et al., 2004; Gregor & Hevner, 2013). The project follows Hevner's (2004) three-cycle framework to execute:

- The Relevance Cycle links instructional practice design activities to using publisher slide decks alongside faculty adaptation challenges.

- The Design Cycle engages in iterative artifact development and refinement through a structured process that combines ChatGPT prompt engineering with a Python automation script to produce updated slides.
- The Rigor Cycle integrates established multimedia learning theories, slide design principles, and instructional technology research to develop the artifact's structure and objectives.

The artifact functions as a reusable modular workflow that includes:

1. Design prompts directing ChatGPT to modify slides to match standard presentation guidelines, including minimal text, visual clarity, and speaker notes.
2. A structured table containing slide metadata: title, revised bullets (semicolon-separated), and speaker notes.
3. A Python script reads slide metadata from an Excel table and creates a PowerPoint deck via the `python-pptx` library.

In a senior IS capstone course, students used a Python script to transform an original publisher-provided slide deck for class presentations. The evaluation of artifact utility depended on its ability to:

- Consistently generate presentation-ready slides
- Save instructor preparation time
- Improve slide alignment with multimedia learning principles

The DSR method's iterative nature, based on early feedback from implementation, enabled the ongoing refinement of both the prompts and the Python code. Due to its transferable design, the artifact requires minimal technical setup yet remains adaptable across various courses or institutions. Following DSR principles requires producing functional solutions that are theoretically grounded and practice-tested while generating insights for future AI-assisted instructional design work.

Artifact Description

This study focuses on a modular workflow that allows educators to apply generative AI and lightweight scripting to redesign textbook PowerPoint slides. The artifact evolved through repeated refinement and functions as both a method and a toolkit to convert text-heavy instructional materials into presentations that follow current educational principles.

The artifact consists of three elements that work together as its fundamental structure. A collection of organized ChatGPT prompts directs the AI to convert a classic slide deck into a concise instructional presentation format. Presentation best practices and cognitive load theory inform these prompts to generate outputs with concise bullet points and minimized on-screen text, while ensuring notes are directed toward the speakers. The AI receives input from instructors, who provide the original slide deck, followed by a series of curated prompts that start with a design guideline primer and end with a request to create a structured slide table.

Figure 1 illustrates the prompt that outlines guidelines for creating slides based on best practices. After that, Figure 2 shows the prompt where the original slide deck is uploaded into ChatGPT, which currently requires a paid subscription.

Issues in Information Systems

Volume 26, Issue 2, pp. 244-257, 2025

Please use the following recommendations when redesigning a slide deck:

Slide Content

- 3–5 bullet points per slide (ideal)
- No more than 6–7 words per line
- Avoid full sentences; use short, key phrases
- Emphasize visual anchors—avoid dense text
- Include images, diagrams, or icons when helpful

Speaker Role

- Slides are prompts—not scripts
- Speaker notes should include:
 - Real-world examples
 - Student stories or case tie-ins
 - Short narratives or open-ended questions
- The goal: encourage engagement, not just reading

Visual Hierarchy

- Use consistent fonts and sizes
- Keep whitespace—avoid clutter
- Use bold/color sparingly for emphasis

Cognitive Load Principle

- Simplify visuals for retention
- Avoid excessive transitions or animations
- One idea per slide is best

Wait for the next prompt before doing anything.

Acknowledged. Slide design principles have been loaded and will guide all revisions going forward.

Let me know when you're ready for the next step (Prompt 2 – Slide Cleanup). I'll wait for your cue.

Figure 1. Prompt 1 Slide Design Guidelines

"I have uploaded a PowerPoint slide deck from a textbook publisher. It does not follow modern presentation design. There are too many complete sentences and cluttered slides. Please analyze and revise it based on best practices."

Wait for the next prompt before doing anything.

Got it. I've reviewed the uploaded slide deck and will apply the best practices from Prompt 1 once instructed.

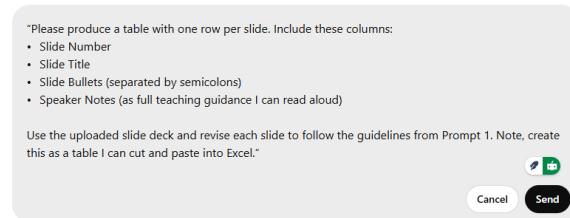
Let me know when you're ready for **Prompt 3 – Create Table**, and I'll start transforming the content into a structured table format for Excel.

Figure 2. Prompt 2 - Modern Presentation Design

The next component is the intermediary representation, shown in Figure 3. The intermediary representation consists of an Excel spreadsheet arranged in a tabular format, which connects AI outputs with the final presentation materials. The table features a row for every slide containing the slide number, title, revised bullet points separated by semicolons to enable parsing, and speaker notes. This standardized format makes AI-generated revisions suitable for automated processing systems.

Issues in Information Systems

Volume 26, Issue 2, pp. 244-257, 2025



The image shows a Microsoft Excel table titled "Revised Slide Table". The table has four columns: "Slide Number", "Slide Title", "Slide Bullets", and "Speaker Notes". The data is as follows:

Slide Number	Slide Title	Slide Bullets	Speaker Notes
1 1	Chapter 9 Digital Value Capture		Use this slide to engage student: with examples of stories. Ask: How does this apply in
2 2	Learning objectives	To analyze the potential of IT-dependent strategic initiatives; To help you Creating added value is just one	Use this slide to engage student: with examples of stories. Ask: How does this apply in Use this slide to engage student: with examples of stories. Ask: How does this apply in

Here is your revised slide table, structured for Excel and formatted using the slide design guidelines from Prompt 1. Each row includes the slide number, title, simplified bullet points (semicolon-separated), and speaker notes written for classroom delivery.

Let me know when you're ready for Prompt 4 – Create a Table, and I'll generate the Python script that

Figure 3. Prompt 3 - Create a Table

The third element in the system consists of a Python script that processes the structured Excel file to create a PowerPoint presentation deck. The script, developed using the python-pptx library, maintains a consistent layout style (Title and Content), imports bullets as individual paragraphs, and places speaker notes below each slide. The automation eliminates the need for manual slide editing, significantly reducing instructor workload while maintaining consistent formatting. These are shown in Figure 4 and Figure 5.

It includes columns:
• Slide Number
• Slide Title
• Slide Bullets
• Speaker Notes

Please write a Python script that:
• Reads Input.xlsx
• Creates a slide deck using a Title and Content layout
• Splits bullet points at semicolons
• Adds speaker notes beneath each slide

Here's the Python script that fulfills your request. This script reads from `Input.xlsx`, processes each row, and generates a PowerPoint slide deck with a Title and Content layout. Bullet points are split at semicolons, and speaker notes are added under each slide.

```
python
from pptx import Presentation
from pptx.util import Inches
import pandas as pd

# Load the Excel file
df = pd.read_excel("Input.xlsx")

# Create a new PowerPoint presentation
prs = Presentation()

# Define standard slide layout (Title and Content)
layout = prs.slide_layouts[1] # Title and Content

for _, row in df.iterrows():
    slide = prs.slides.add_slide(layout)

    # Set slide title
    slide.shapes.title.text = str(row["Slide Title"])

    # Set slide content bullets
    content = slide.placeholders[1]
    bullet_points = str(row["Slide Bullets"]).split(";")
    content.text = bullet_points[0].strip() if bullet_points else ""
    for bullet in bullet_points[1:]:
        p = content.text_frame.add_paragraph()
        p.text = bullet.strip()

    # Add speaker notes
    notes_slide = slide.notes_slide
    notes_slide.notes_text_frame.text = str(row["Speaker Notes"])

# Save the presentation
prs.save("Generated_Slides.pptx")
print("Presentation generated successfully: Generated_Slides.pptx")
```

Steps to run:

1. Save this code to a `.py` file (e.g. `generate_slides.py`).
2. Ensure your `Input.xlsx` file is in the same directory.
3. Open a terminal and run:

```
bash
python generate_slides.py
```

Let me know if you'd like me to run this on the table we just created or help format it for download.

for _, row in df.iterrows():
slide = prs.slides.add_slide(layout)

Set slide title
slide.shapes.title.text = str(row["Slide Title"])

Set slide content bullets
content = slide.placeholders[1]
bullet_points = str(row["Slide Bullets"]).split(";")
content.text = bullet_points[0].strip() if bullet_points else ""
for bullet in bullet_points[1]:
p = content.text_frame.add_paragraph()
p.text = bullet.strip()

Add speaker notes
notes_slide = slide.notes_slide
notes_slide.notes_text_frame.text = str(row["Speaker Notes"])

Save the presentation
prs.save("Generated_Slides.pptx")
print("Presentation generated successfully: Generated_Slides.pptx")

Steps to run:

1. Save this code to a `.py` file (e.g. `generate_slides.py`).
2. Ensure your `Input.xlsx` file is in the same directory.
3. Open a terminal and run:

```
bash
python generate_slides.py
```

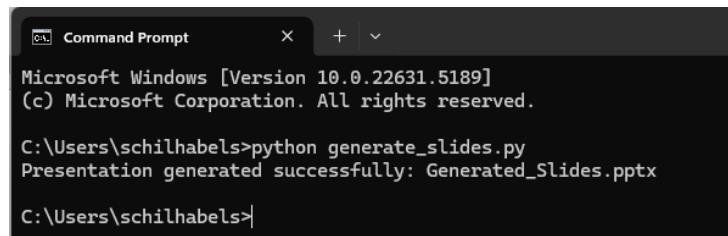
Let me know if you'd like me to run this on the table we just created or help format it for download.

Figure 4. Prompt 4 - Python Script

Figure 5. Prompt 4 – (Continued)

248

The creation of this artifact focused on both accessibility and transferability features. A simple Python environment and file-enabled ChatGPT access, including ChatGPT Plus with GPT-4 4 are the only prerequisites needed. Once set up, the workflow becomes reusable for different courses and instructors. Through this process, faculty members can achieve improved slide quality and gain a practical demonstration of embedding generative AI into their workflows, leading to better instructional design. Figure 6 shows the actual command prompt to run the Python script. No programming knowledge is required. ChatGPT generates the code; the user needs to execute it. If there are errors, take a screenshot and feed it back into ChatGPT for analysis.



```
Command Prompt
Microsoft Windows [Version 10.0.22631.5189]
(c) Microsoft Corporation. All rights reserved.

C:\Users\schilhabels>python generate_slides.py
Presentation generated successfully: Generated_Slides.pptx

C:\Users\schilhabels>
```

Figure 6. Python Execution Command

Demonstration and Application

The workflow demonstrated the functionality and instructional utility of the artifact by applying it to a senior-level Information Systems (IS) capstone class at the University of Wisconsin Oshkosh. The slide deck from Chapter 9 of Information Systems for Managers (Piccoli & Pigni, 2021) served as the source material from a widely used textbook that includes PowerPoint slides provided by the publisher. The original slide decks shared characteristics with other decks, which combined lengthy paragraphs and complete sentences with visuals that did not match current presentation norms.

The original slide deck was uploaded to ChatGPT using the developed prompts and the advanced capabilities of GPT-4, which included file upload functionality. The instructor used a series of prompts to create visual and pedagogical standards, analyze the slide deck, and generate a restructured table of contents. The structured table was copied into an Excel worksheet using the artifact's designated format. The Python script was run to transform the structured table into a completely redesigned .pptx presentation. Figure 7 shows the original slide number seven. As you can see, there are bullets, but the format contains nearly complete sentences.

Resource-Based View (RBV)

- A firm is modeled as a bundle of resources → firm's competitive advantage depends on the characteristics of the resources at its disposal, and when the firm controls resources that are:
 - Valuable: it underpins a value-adding strategy
 - Rare: it is idiosyncratically distributed
 - Inimitable: it is impossible, or difficult, to duplicate for competitors
 - Nonsubstitutable: competitors are unable to replicate the firm's overall value proposition using surrogate resources for the ones that are rare, valuable, and inimitable
- In this case the advantage will be difficult for competitors to overcome

Figure 7. Original publisher-provided slide.

Source: Piccoli & Pigni, IS for Managers, Edition 5.0, © 2022. Used under fair use for instructional and scholarly purposes.

The updated slide deck contained 29 individual slides. Reformatting resulted in slides featuring three to five bullet points that did not exceed 6–7 words in length. Figure 8 shows the ChatGPT revision and contains the updated slide seven. Speaker notes were integrated to serve as contextual prompts for instructors, along with narrative components and cues for classroom interaction. During the live class session, instructors used the updated presentation deck instead of the original textbook-provided materials.

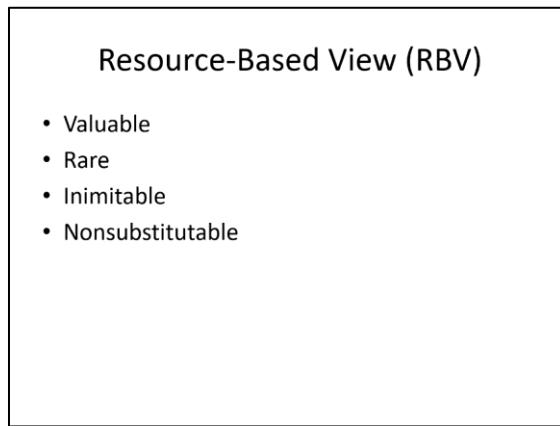


Figure 8. ChatGPT Generated Slide

Figure 9 shows the new slide seven with two manual changes. First, the UW Oshkosh PowerPoint theme was applied. To make the slide more visually appealing, the Smart Art feature of PowerPoint was used to create a visual representation. Figure 10 shows the speaking notes that were added. Speaker notes were integrated to serve as contextual prompts for instructors, along with narrative components and cues for classroom interaction. This assumes that the instructor is familiar with the concepts being presented. If this were not the case, the prompt could be modified to have ChatGPT increase the speaker notes to include full definitions and/or examples or antecedents for the speaker to reference. During the live class session, instructors used the updated presentation deck instead of the original textbook-provided materials.

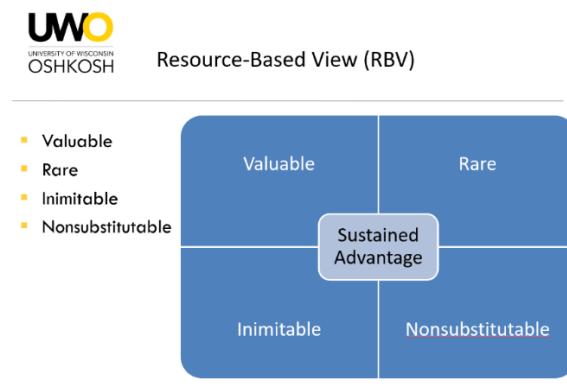


Figure 9. Final Slide Seven Edited by Instructor (cut and paste for visualization)

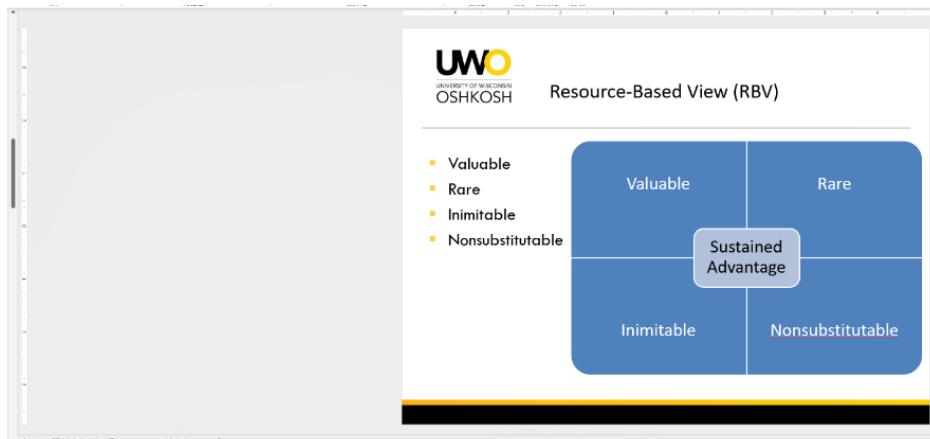


Figure 10. Slide seven with speaking notes

The workflow application finished within sixty minutes and incorporated AI interaction, content review, and slide generation. In earlier semesters, manual editing of similar slide presentations generally occupied numerous hours of instructor labor. Redesigned slides received positive feedback from follow-up discussions and peer review because they were more engaging and educationally effective.

The demonstration illustrates how educators can seamlessly integrate the artifact into their teaching workflows with minimal technical configuration, while producing high-quality materials that can be easily reused. Combining AI and automated systems with instructional design activities demonstrates how generative tools boost faculty productivity while maintaining high instructional standards.

Evaluation

The artifact was evaluated through its application in a real instructional setting and assessed along several key dimensions: The evaluation metrics included design principles alignment, time efficiency, consistent output delivery, and instructional usefulness. The evaluation investigated the artifact's technical performance and its educational effectiveness within classroom settings.

Comparison of Original vs. Revised Slides

The best evidence of the artifact's effectiveness comes from visually contrasting the original publisher-provided slide with its revised version created by the workflow. The initial slide presentation (Figure 6) contained extensive paragraphs of text with no visual prioritization and failed to provide adequate tools for classroom interaction. In contrast, the revised slide (Figure 8) adheres to established presentation design principles. The slide revision incorporates brief bullet points while maintaining a strict word count and includes notes for the instructor's benefit. The side-by-side comparison demonstrates how transforming dense text into short bullet points reduces cognitive load. The slide redesign enables instructors to elaborate verbally on each concept, resulting in improved audience involvement and interaction.

Time and Consistency

The process of creating the redesigned deck used AI interaction, automated deck generation, and table cleanup, and was completed in under one hour. Manual slide revision by instructors used to take three to five hours for each chapter. The time savings reached their peak in courses with multiple sections or rotating instructors, as the artifact could be reused with only slight modifications.

A standard prompt set combined with a structured input table-maintained consistency throughout the slides and decks. The slides maintained a predictable format, bullets kept a consistent style, and speaker notes linked automatically. The standardized design eliminated the need for additional editing and allowed for straightforward integration with supplementary instructional content.

Instructional Utility

According to instructor feedback, live instruction with revised slides demonstrated enhanced clarity and pacing. The updated format enhanced storytelling while enabling spontaneous examples and permitting more flexible discussions. The students' feedback showed they found the new slides simpler to understand and less daunting. Presenters new to teaching found the speaker notes especially useful because they delivered just-in-time support during their presentations. The notes allowed presenters to keep their presentations seamless by avoiding detailed scripting for each slide, which improved their teaching confidence.

Lessons Learned

"Success" was the ability to generate reliable, pedagogically viable, and time-efficient teaching material that fostered interest and understanding. It was used in seven course sections over two semesters, delivered in weekly blocks during 14-week terms, for a total of 98 class sessions. Actual classroom implementation served as evidence of educational value and sustainability. Student instructor observation, time log, and informal student feedback were used for evaluation. Positive improvements in instructional clarity and efficiency, with limitations in user accessibility and AI configuration dependencies. Future iterations will explore technical configuration barriers and incorporate multimedia support.

Limitations and Challenges

The workflow produces positive results, but it operates within certain constraints. The process requires using a file-enabled ChatGPT version, such as ChatGPT Plus, with GPT-4, which may not be available to everyone. Instructors need to install and configure Python, which could pose a modest technical obstacle for some users. The current workflow lacks support for image embedding and advanced formatting features, which restricts its effectiveness for disciplines that rely heavily on visual elements. The workflow offers significant automation combined with evidence-based practices and measurable time efficiency, making it an essential tool for educators who want to enhance their teaching materials with minimal work.

Results

It was implemented in seven sections over 12 months and integrated into the weekly schedule in all 14-week sessions. The effectiveness of the system was evaluated in terms of the following four categories: collaboration and communication, task visibility and responsibility, tool adoption and opposition, and instructional efficiency and repurposing.

Collaboration and Communication

Utilizing Teams as a centralized platform for discussion and organization improved communication between sections and reduced reliance on email and bulletin board-style announcements. Discussion channels allowed threaded posts, shared file storage, and structured threads to enhance communication transparency for both students and teaching assistants. Instructors were able to communicate more directly

and streamline the messaging process while maintaining a constant history of communications that was accessible to all members of the section.

Task Visibility and Responsibility

The planner allowed each module of the course to be broken down into individual tasks, each with its due dates, tags, and owners. This increased visibility, of course, benefits both instructors and students. Taskwork was externalized by the system, reducing confusion about deadlines and assignments. The course tended to have fewer late submissions and fewer inquiries about “what is due” each week over the semester.

Tool Adoption and Opposition

Resistance to the platform tended to occur from students who did not have experience with Teams or had previously only used the institution’s platform (Canvas). Adoption was built up over the semester through the continual utilization of the platform in conjunction with weekly tasks. Once students engaged with Teams, they reported greater clarity and less anxiety about the course structure.

Instructional Efficiency and Repurposing

The structured nature of the course allowed for the reuse of instructional materials (slides, lists, handouts, etc.) across multiple sections with minimal modification. This provided considerable time savings for multi-section courses. The effort for instructors was shifted away from administrative work to focus on pedagogical interactions with students, providing opportunities for active teaching and feedback.

Discussion

Through the analysis of this artifact, we see that generative AI systems become valuable tools in instructional design when combined with thoughtful prompt engineering and consistent workflow implementation, especially for tasks that demand significant effort but offer little creative reward, such as slide deck revision. The research highlights how AI functions as a collaborative design tool in educational settings, offering a practical roadmap for educators to integrate automated systems into their instructional preparation work.

Based on design science criteria, this artifact matches the guiding principles of relevance, rigor, and utility. The workflow was designed to address a highly relevant challenge faced by instructors: The process of modifying textbook slide decks creates both significant time demands and cognitive workload for instructors. Researchers aligned the artifact with cognitive load theory and visual design principles, alongside multimedia learning standards, to establish academic rigor and tested its effectiveness through real classroom applications. The system demonstrates utility by producing measurable time savings, consistent output quality, and positive instructional feedback.

The artifact provides time savings and improved slide clarity benefits, enabling a wider discussion about faculty empowerment during AI integration. This workflow design frames ChatGPT as a supporting generative assistant, allowing instructors to maintain authority over teaching choices while delegating routine formatting and restructuring tasks to the AI. AI functions as a partner to improve teaching methods rather than serving as a mere gimmick. The artifact facilitates consistent teaching methods across different courses while ensuring equal treatment for all students. A standardized redesign workflow in multi-section or team-taught courses delivers high-quality visual materials to all students, irrespective of which instructor teaches them. The artifact helps newer faculty members teach complex subjects effectively by reducing barriers to teaching.

The artifact requires additional refinement because generative AI capabilities continue to advance. This version lacks visual media production capabilities and interactive slide features and does not connect with institutional learning management systems. Future research and development opportunities emerge from these proposed enhancements. Most importantly, this project highlights a subtle yet powerful shift: educators can now achieve educational excellence without compromising operational efficiency. Precision-designed tools enable the simultaneous enhancement of pedagogical quality and efficiency, scaling benefits throughout courses, programs, and institutions.

Though this artifact was developed in the context of one faculty member's classroom practice, its underlying design principles (use of AI tools to guide slide development, structured question prompts, and ready-to-use templates) could be generalized to other disciplines, whether business, health sciences, education, or even the humanities. Because Microsoft Teams, ChatGPT, and Python-based coding tools are all platform-agnostic, they could be leveraged institution-wide without customization. They could easily scale across multiple departments, where teachers might otherwise need to adapt this method to the field of instruction (e.g., for use in humanities classes, the department chair would need to modify the input table structure, but content expectations could remain largely the same). Adjunct or rotating instructors could also benefit from the standardized form and the incorporation of speaker notes.

Research Contribution

This project is an example of a DSR artifact that was initially created and iteratively improved through a real-life deployment over seven university-level courses. The contribution includes the artifact itself, a reproducible, AI-enhanced workflow for instructional slide redesign, as well as knowledge of the process that creates it, specifically, how to integrate generative AI tools (i.e., ChatGPT, Python scripts) into instructional design. The goal of this work is to provide a practical, low-cost solution that enhances the clarity, consistency, and efficiency of content presentation while also demonstrating a proof-of-concept for connecting collaborative software platforms (e.g., Microsoft Teams) with AI-driven instructional support systems. This research contributes to the IS research literature related to technology-assisted pedagogy and the integration of collaboration tools in education.

Conclusion and Future Work

This paper presented a design science artifact that addresses a widespread but often overlooked challenge in higher education: Textbook-provided slide decks exhibit both inefficiency and pedagogical weaknesses in their current form. Integrating ChatGPT's generative power with Python-based automation enables this artifact to deliver a scalable solution that transforms text-heavy slides into visually appealing instructional materials, adhering to modern design principles. Implementing the workflow in a senior-level IS course substantially reduced instructor preparation time and enhanced the quality of visual and pedagogical materials. The newly structured decks helped reduce cognitive load and deliver narratives while maintaining instructional uniformity across sessions. The tool proved suitable for wider implementation in various academic fields and institutions, particularly when instructors rely on standardized textbooks and have restrictions on both time and instructional design help.

The project advances instructional practice and design science literature by developing a theoretical artifact from contextual creation and assessment in practical application. This work highlights AI's dual function as both a creator of content and an active participant in designing instructional tools that address current educational needs.

The path forward includes several distinct areas that warrant future research. The workflow should be enhanced to support visual media, allowing the AI system to suggest or add appropriate icons, diagrams, or illustrations. Integrating learning management systems (LMS) would simplify the distribution of finalized slides while ensuring their alignment with educational modules and assessments. Creating a prompt template library with reusable input structures can help promote faculty adoption across several departments.

Implementing this workflow enables a wider exploration of AI-assisted instructional design research, which encompasses topics such as co-authorship, instructional transparency, and ethics of teaching automation. Through ongoing exploration of generative tools, educators can utilize projects like this as examples for integrating human understanding with machine power to enhance education quality and accessibility.

References

Duarte, N. (2008). *Slide:ology*. O'Reilly Media.

Gregor, S., & Hevner, A. R. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly*, 37(2), 337–355.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1), 75–105.

Mayer, R. E. (2009). *Multimedia learning* (2nd ed.). Cambridge University Press.

Mollick, E. R., & Mollick, L. (2023). Using AI to implement effective teaching strategies in classrooms: Five strategies, including prompts (Wharton School Research Paper). SSRN. <https://doi.org/10.2139/ssrn.4391243>

Peffers, K., Rothenberger, M., Tuunanen, T., & Vaezi, R. (2012). Design science research evaluation. In K. Peffers, M. Rothenberger, & B. Kuechler (Eds.), *Design science research in information systems* (pp. 398–410). Springer.

Piccoli, G., & Pigni, F. (2022). *Information systems for managers* (5th ed.). Prospect Press.

Reynolds, G. (2012). *Presentation zen: Simple ideas on presentation design and delivery* (2nd ed.). New Riders.

Appendix: Technical Setup Guide

Part A: Required Software and Services

- ChatGPT Plus Account
- Microsoft Excel
- Python Programming Environment

Part B: Installing Python

- Download the Python installer: Visit <https://www.python.org> and choose Python 3.10 or higher.
- Run the installer: Check 'Add Python to PATH' and click 'Install Now'.

Note: You may need administrative privileges. Contact your IT support if needed.

Part C: Installing Required Python Libraries

- Open a terminal or command prompt.
- Run the command: pip install pandas python-pptx openpyxl

Part D: Preparing the Excel Input File

- Create columns: Slide Number, Slide Title, Slide Bullets (semicolon-separated), Speaker Notes
- Each row = one slide; no line breaks in cells.
- Save as Input.xlsx in the script folder.

Part E: Running the Python Script

- Save script as generate_slides.py
- Place it with Input.xlsx in the same folder
- Open terminal, navigate to folder using: cd path_to_your_folder
- Run script: python generate_slides.py

Part F: Common Issues and Troubleshooting

The table below lists the common issues that were found, identifying possible causes and suggested solutions.

Issue	Possible Cause	Suggested Solution
'pip not recognized'	Python was not added to PATH	Reinstall Python and check 'Add to PATH'
'Permission denied'	No admin rights	Contact IT for help
Empty slides generated	Missing bullet delimiter	Use semicolons between bullet points in Excel
Script error	File names or paths are incorrect	Make sure Excel and the script are in the same folder

Part G: ChatGPT Prompt 1 – Slide Design Guidelines

Please use the following recommendations when redesigning a slide deck:

Slide Content:

- 3–5 bullet points per slide (ideal)
- No more than 6–7 words per line
- Avoid full sentences; use short, key phrases
- Emphasize visual anchors—avoid dense text
- Include images, diagrams, or icons when helpful

Speaker Role:

- Slides are prompts, not scripts
- Speaker notes should include:
 - Real-world examples
 - Student stories or case tie-ins
 - Short narratives or open-ended questions
- The goal: encourage engagement, not just reading

Visual Hierarchy:

- Use consistent fonts and sizes
- Keep whitespace—avoid clutter
- Use bold/color sparingly for emphasis

Cognitive Load Principle:

- Simplify visuals for retention, one idea per slide is best
- Avoid excessive transitions or animations

Part H: ChatGPT Prompt 2 – Slide Deck Cleanup

“I have uploaded a PowerPoint slide deck from a textbook publisher. It does not follow modern presentation design. There are too many complete sentences and cluttered slides. Please analyze and revise it based on best practices.”

Part I: ChatGPT Prompt 3 – Create Table

“Please produce a table with one row per slide. Include these columns:

- Slide Number
- Slide Title
- Slide Bullets (separated by semicolons)
- Speaker Notes (as full teaching guidance, I can read aloud)

Use the uploaded slide deck and revise each slide to follow the guidelines from Prompt 1. Note, create this as a table I can cut and paste into Excel.

Part J: ChatGPT Prompt 4 – Create Slides

“I have pasted the slide table into an Excel file called Input.xlsx.

It includes columns:

- Slide Number
- Slide Title
- Slide Bullets
- Speaker Notes

Please write a Python script that:

- Reads Input.xlsx
- Creates a slide deck using a Title and Content layout
- Splits bullet points at semicolons
- Adds speaker notes beneath each slide.”