



MIS 112 Project Instructions, Rubrics, and Sample Projects Capstone Project

Design, build, evaluate, and communicate a substantive, semester-long generative-AI project that demonstrates deep understanding of an AI problem, thoughtful method choice, rigorous evaluation, and responsible use.

Specific Suggestions:

1. Use AI tools to brainstorm, prototype, and test your project—capture the prompts and outputs you used.
2. Be transparent: clearly label what the AI produced, include the model name/version and prompt history, and show the concrete steps you took to validate results.
3. If a sample project or instruction is unclear, ask an AI tool for clarifying examples or follow-up questions and include the AI's suggestions.
4. Choose a project that centers on hands-on model use (prompting, iterative refinement, agent sessions, multimodal generation, RAG, etc.), not a passive summary.
5. If you need help, contact us with your prompt logs and example outputs so we can provide targeted guidance.

What you may build: tool or prototype (no-code or code), empirical research / benchmark, case study or business use-case, process automation design, or a proof-of-concept exploration. Analysis/evaluation projects are fully acceptable — coding is **not** required.

Deliverables:

1. **Project Report (pdf, 8-12 pages):**
 - i. Clear problem statement, objectives, related work,
 - ii. Methods (tools/models),
 - iii. Experimental design,
 - iv. Results,
 - v. Interpretation and evaluation,
 - vi. Ethics/limitations section, and
 - vii. Conclusion

2. **Appendix (required):** prompts / prompt version history, tool names + versions, links/screenshots, dataset descriptions (source, license, sampling), evaluation rubric or metrics, and any Excel sheets or exported logs.
3. **Presentation slides (5-8 slides)** for in-class project synthesis.

Technical depth expectations:

- You are **not** required to write code. Rigor is shown by: careful experimental design, controlled comparisons, reproducible prompt chains or tool configurations, and meaningful evaluation (quantitative or qualitative).
- If you *do* implement (no-code or light code), include reproducibility notes: steps to reproduce, exact prompts, tool versions, and any parameters.

Ethics & responsible use (required):

- Disclose AI assistance and indicate what parts were AI-generated.
- Do **not** upload personal/identifiable data. Use public / synthetic / instructor-approved datasets only.
- Include a short **Ethics Checklist** in your appendix addressing privacy, consent (if applicable), copyright, bias assessment, potential misuse, and a mitigation plan. See mandatory checklist at the end of this packet.

Submission format & naming: Upload to D2L, a single ZIP file or PDF document named “Lastname_ProjectTitle” containing the report, appendix files, slides, and demo link/file.

Grading Rubric

100 points (five equally weighted criteria; 20 points each)

Criterion A – Problem Definition & Relevance (20 pts)

What is measured: clarity of problem statement, explicit objectives, scope (narrow and achievable), stakeholder or user value, and alignment between goals and chosen evaluation measures.

Criterion B – Selection & Correct Application of AI Methods (20 pts)

What is measured: appropriateness of chosen generative-AI techniques or tools, justification for choices, correct conceptual application (e.g., why retrieval + LLM or why few-shot prompts), and linkage to evaluation.

Criterion C – Technical Rigor, Evaluation & Reproducibility (20 pts)

What is measured: experimental design or evaluation strategy quality (controls, baselines, metrics), clarity of results, statistical or qualitative rigor, reproducibility (prompts, versions, data sources), and honest reporting (failed experiments included).

Criterion D – Innovation, Integration & Practical Value (20 pts)

What is measured: creativity of solution, effective integration of AI into a coherent product or analysis, user-centered considerations (usability, deployment plan, handoff to humans), and real-world usefulness or insight.

Criterion E – Ethics, Limitations & Documentation (20 pts)

What is measured: explicit analysis of limitations/biases/harm, mitigation strategies, proper citations/licenses for data/models, full documentation (prompts, versions, datasets), and clear disclosure of AI assistance.

Sample Projects

Here are sample projects that would be appropriate for this course. You do not need to pick for this list, and it would be better if you used this list as an idea generator and customized the project to your own interests. Also feel free to do a project that is completely unrelated to this list.

These projects are related to the topics we've covered in this course and use the following assumptions:

- No assumed background beyond course topics
- HP Elitebook x360 830 G8 running Windows 11 with Chrome
- No personal data uploads
- No coding
- Software only MS Office, e.g., Word, Excel, etc. (if needed)
- All tasks run in the browser; no installs; no admin rights required

Topic 1 – Foundations of Machine Learning & LLM behavior

1. **Token Continuation Lab** — Iteratively prompt an LLM to continue the same three-paragraph seed text across 10 prompt variants; produce the 10 continuations and a short annotated prompt log showing how prompt changes alter coherence and style.
2. **Hallucination Finder** — Give factual queries to three browser LLMs, collect outputs, and create an annotated error catalogue (example + model output) highlighting hallucination types and reproducible prompt triggers.
3. **Context Window Stress Test** — Feed progressively longer context chunks to a multimodal playground and observe where context is lost; deliver side-by-side screenshots of inputs and outputs showing the breakdown points.
4. **Token-Efficiency Challenge** — Solve a fixed task (e.g., translate + summarize) under a strict token budget using prompt compression strategies; students submit the final compressed prompt and model output.
5. **Interactive Classifier with Teachable Demos** — Build a no-code image/text classifier using a browser demo (Teachable Machine or equivalent), train with supplied samples, and demo live classification on new examples.

Topic 2 – Generative Text: Prompting & Output Control

6. **Few-Shot vs Zero-Shot Comparison** — For the same complex task, craft zero-shot and 3-example few-shot prompts; run both on two models and submit prompt-output pairs with short example-based commentary.
7. **Style Transfer Chain** — Use an LLM to convert one paragraph into five distinct styles (formal, casual, persuasive, technical, poetic) and assemble a one-page style-gallery with each prompt and output.
8. **Iterative Revision Workshop** — Start with a low-quality LLM answer and use only model prompts (no manual edits) to iteratively improve it through 4 refinement steps; submit the revision chain.
9. **Constraint Prompting Exercise** — Instruct a model to obey a tight constraint set (word limits, banned terms, tone) and test with adversarial prompts to see constraint robustness — deliver transcripts showing failures and successes.
10. **Prompt-engineered Q&A Bot** — Build a small Q&A flow in a web playground (no-code) that uses system and user prompts to guide tone and factuality; demonstrate with a 5-question live session.

Topic 3 – Retrieval-Augmented Generation (RAG) & Embeddings

11. **Mini RAG Build** — Upload a short class document to a browser RAG demo, create embeddings, and run 10 retrieval questions; deliver the RAG vs plain LLM answer pairs and the retrieval logs.
12. **Search Sensitivity Test** — Vary retrieval chunk size and observe answer changes for the same question; submit examples that show when retrieval helps or hurts.
13. **Embedding Neighborhood Exploration** — Create embeddings for 20 sample sentences, visualize nearest neighbors via screenshots, and annotate surprising clusters discovered.
14. **Source Attribution Challenge** — Use a RAG playground to answer assertion-heavy prompts and test citation accuracy across 8 queries, collecting prompt-output-citation triplets.
15. **Interactive FAQ RAG Agent** — Configure a no-code agent that uses the uploaded FAQ doc to answer questions, then conduct a live admin demo session showing how the agent sources its answers.

Topic 4 – AI Agents & Workflows

16. **Study-Plan Agent** — Assemble a no-code agent that converts an assignment checklist into a minute-by-minute study plan; run three different assignment scenarios and submit agent logs.
17. **Task Routing Agent** — Build an agent that decides whether to use web search vs. knowledge-base retrieval for each incoming question; demonstrate with 10 mixed queries and show routing decisions.
18. **Dialog Policy Tester** — Create an agent with explicit guardrails (safety prompts) and produce transcripts that test guardrail robustness with adversarial prompts.
19. **Agent-as-Assistant Demo** — Use a web agent builder to create a "meeting summarizer" that ingests meeting notes and outputs action items; demo on three sample meeting notes.

- 20. Interactive Troubleshooter Agent** — Build a decision-tree agent for a common task (e.g., printer troubleshooting) and run live user-testing sessions to gather improvement notes.

Topic 5 – Image Generation & Visual Prompting

21. **Cover Art Generator** — Use a browser image generator to produce five cover-image variants from a one-line story prompt; deliver the images with the exact prompts that produced each.
22. **Style Prompt Calibration** — Keep content fixed and systematically vary style instructions (photoreal, oil painting, low-poly) to show how style keywords affect composition; submit image grid.
23. **Image Forensics Play** — Generate images then run detection tools in demos to check detectability; present side-by-side pairs (image + detection output) and concrete observations.
24. **Accessible Alt-Text Designer** — Feed images to a multimodal model to produce alt-text, then refine via iterative prompts until alt-text passes an accessibility checklist; deliver final alt-text plus prompt log.
25. **Multimodal Storyboard** — Create six storyboard panels for a 60-second scene using an image model plus short descriptive captions produced by an LLM in-browser.

Topic 6 – Audio Generation & Speech

26. **Mood Intro Composer** — Use a browser audio generator to produce three 30-second podcast intro tracks for different moods; submit audio files and the prompts used.
27. **Speech-to-Text Error Audit** — Transcribe three provided sample audios with a web ASR demo, correct transcripts, and submit WER calculations plus examples of typical errors.
28. **TTS Persona Builder** — Assemble a short branded voice intro using a demo TTS tool (allowed demo voices only), iterate on prompt/SSML to refine prosody, and deliver final audio plus prompt parameters.
29. **Audio Style Transfer** — Convert a short melody prompt into three stylistic interpretations using audio generation demos and present the waveform screenshots and audio links.
30. **Podcast Q&A Generator** — Use an LLM to build a Q&A script for a short mock interview, then synthesize both interviewer and guest voices using TTS and deliver the audio dialogue.

Topic 7 – Computer Vision & Multimodal Reasoning

31. **Object Detection Walkthrough** — Run an object detection demo on class-supplied images, capture predictions, and create a confusion snapshot showing false positives/negatives.
32. **Image + Text Q&A** — Use a multimodal model to answer detailed questions about an image plus an attached short passage; deliver Q&A transcripts demonstrating multimodal grounding.

33. **Accessibility Captioning Flow** — Build a small pipeline in a browser tool that converts images to descriptive captions and short summaries for visually impaired users; demo with 6 images.
34. **Video Frame Analysis** — Use a web demo that extracts and labels frames from a short sample video; present labeled frames and discuss recognition gaps.
35. **Visual Reasoning Puzzle** — Present a logic puzzle embedded in an image and use a multimodal model to solve it step-by-step, submitting the model's reasoning transcript.

Topic 8 – Prompt Engineering: System & Chain-of-Thought

36. **Chain-of-Thought Elicitation** — Design prompts that encourage step-by-step reasoning for a multi-step finance or planning task and capture the chain-of-thought outputs across 5 prompts.
37. **Instruction Precision Trial** — For a complex instruction, produce 4 prompt variants with increasing specificity and compare how the answer precision changes in model outputs.
38. **Prompt Library Creation** — Build a short, reusable library of 8 prompts for a common task (resume rewrite, email tone polish, etc.), test each with an LLM, and deliver the prompt bank plus live examples.
39. **A/B Prompt Competitions** — Run head-to-head A/B tests of two prompt strategies on the same evaluation set and submit anonymized voting logs from peer testers showing preferences.
40. **Adversarial Prompting Defense** — Create adversarial prompts designed to break model constraints and then craft defensive system prompts to block them; demonstrate before/after transcripts.

Topic 9 – No-Code Chatbots & Conversational UX

41. **FAQ Chatbot Build** — Using a browser bot-builder, create an FAQ chatbot for a supplied fictional service, run 10 live user queries, and deliver the full chat transcript plus improvement notes.
42. **Tone-Tuning Chatbot** — Build the same chatbot with three distinct tones (formal, friendly, terse) and demo the tone differences via identical user prompts.
43. **Escalation Flow Designer** — Construct chat flows that detect when to escalate to human help (simulated) and demonstrate the detection on 12 sample chats.
44. **Persona-Based Tutor Bot** — Make a tutor chatbot persona that teaches a short skill; run peer testing and submit recorded sessions demonstrating instruction clarity.
45. **User Simulation Testing** — Use an LLM to simulate 50 varied user inputs and run them through your chatbot to find weaknesses; deliver a prioritized bug list with transcript examples.

Topic 10 – Business Applications & Prototyping (artifact-focused)

46. **Ad Copy Generator & A/B Files** — Use an LLM to create 10 ad variants for a supplied product brief, then run a blind peer preference test and submit ranked outputs and demo prompts.

47. **Pitch Slide Text Generator** — Produce the text and speaker notes for a five-slide pitch using an LLM, then convert into a slide deck mockup using a browser slide editor (no report).
48. **Customer Reply Generator** — Build an agent that drafts customer support replies from ticket text; validate by running it against 20 sample tickets and deliver the top 10 improved drafts.
49. **Expense Categorizer Agent** — Configure a model to classify sample expense lines into categories and reveal the confusion cases with sample inputs.
50. **Market Persona Creator** — Generate three buyer personas (text + attributes) from a short product description using an LLM and present them as interactive persona-cards.

Topic 11 – Creative Generative Projects

51. **Interactive Fiction Prototype** — Create a branching text-adventure using an LLM-powered web tool; publish a playable 10-turn demo and provide play logs.
52. **Album Art + Track Snippet** — Generate cover art and a short instrumental audio snippet for a fictional single using image + audio generators and deliver both assets.
53. **Poetry Voice Series** — Generate eight poems in a consistent poetic voice using iterative prompting and present as a themed digital chapbook (images + poems).
54. **Logo Iteration Workshop** — Use an image model to produce 8 logo concepts; present live critiques and select refinements via additional prompts.
55. **Short Scene Filming Plan** — Use an LLM to generate a 60-second scene screenplay and then produce six AI-generated storyboard images matching the scene.

Topic 12 – Research Practices, Reproducibility & Method Demos

56. **Repro Checklist Generator** — Give a methods paragraph to an LLM and have it generate a step-by-step runnable checklist (no code) that others could follow; test the checklist on a peer to confirm clarity.
57. **Hypothesis-to-Experiment Designer** — Provide a toy dataset and use an LLM to propose three testable hypotheses and concrete experiment steps that can be simulated with supplied tools.
58. **Protocol Visualizer** — Convert a short methods description into a figure generated by an image model that clarifies the workflow; submit the figure and alt text.
59. **Synthetic Data Augmentation** — Use an LLM to generate synthetic short text samples to expand a small dataset, then demonstrate model behavior differences on original vs augmented sets.
60. **Peer Repro Playtest** — Exchange prompt+input with a peer, have them re-run the interaction in the browser, and submit the re-run transcript to show reproducibility.

Topic 13 – Ethics, Bias, and Safety Labs

61. **Bias Audit by Prompting** — Systematically prompt a model with controlled demographic variations and collect outputs to create an evidence file of differential responses (no narrative report).
62. **Privacy Leak Simulation** — Use synthetic identity prompts to test if models reveal private-like content; document exploit prompts and model reactions in a log for mitigation design.
63. **Consent-First Voice Demo** — Create a TTS persona only from provided sample scripts and produce a consent template plus recorded demo showing explicit consent phrasing.
64. **Harm Scenario Roleplay** — Use an LLM to generate mitigation steps for hypothetical misuse scenarios, then roleplay the mitigations in a safety-agent demo session.
65. **Explainability Walkthrough Demo** — Have an LLM produce stepwise plain-English explanations for its own answer to a decision and present the explanation as a short live demo.

Topic 14 – Society, Workflows & Future Design

66. **Human+AI Workflow Prototype** — Use an LLM to simulate the AI half of a human-AI workflow (e.g., draft + human edit) and produce a live before/after demo of task throughput improvements.
67. **Upskilling Micromodule** — Produce a short interactive lesson (chatbot-driven) that teaches one micro-skill; test with peers and present completion logs.
68. **Job Task Reallocation Simulator** — Feed a job description to an LLM and have it produce task reallocations to AI/human cells; present an interactive task matrix artifact.
69. **Community Assistant Prototype** — Build a local-business assistant bot that answers visitor questions and demonstrates scheduling or FAQ handling in-browser.
70. **Policy Brief Generator (artifact only)** — Use an LLM to create a one-page decision checklist for institutional AI adoption, then convert that checklist into an interactive flowchart using a browser editor.