SEIS 666 — SPRING 2026

Class Project

"Build Something That Reasons"

Due: Final class session (Week 14) **Weight:** 30% of final grade

Format: Working system + 10-min presentation

SECTION 1

The Assignment

You will build an AI-powered system that demonstrates structured intelligence — not just prompting, but giving AI the context, relationships, and rules it needs to reason about a real domain.

> This is not a research paper. This is not a slide deck about AI. You are building a working system.

You have three project tracks to choose from:

TRACK A

Knowledge Graph + AI System

Build a knowledge graph for a real domain — your employer, an industry you know, a problem you care about — and demonstrate how structured context changes AI output quality.

DELIVERABLES

- ✓ A domain knowledge graph (minimum 50 entities, 100 relationships)
- ✓ Can use any graph tool: Neo4j, NetworkX, JSON graph, or even structured markdown
- ✓ A working AI integration that consumes the graph (Claude API, ChatGPT, or any LLM)
- ✓ A before/after comparison: same questions answered with and without the graph
- ✓ Documentation of your graph schema (entity types, relationship types, rules)
- ✓ A 10-minute presentation to the class

TRACK B

Agentic AI System

Build a multi-step AI agent that uses tools, makes decisions, and completes a real workflow autonomously.

DELIVERABLES

- ✓ A working agent built with Claude Code, Claude API, OpenAI API, or any agent framework
- ✓ The agent must use at least 2 external tools (APIs, databases, file systems, web scraping, etc.)
- ✓ A documented workflow showing the agent's decision tree / process flow
- ✓ Error handling: what happens when the agent encounters unexpected input?
- ✓ A before/after comparison: how long does this workflow take a human vs. your agent?
- ✓ A 10-minute presentation to the class

TRACK C

Open Track (Instructor Approval Required)

If you have a different idea that demonstrates AI-powered digital transformation at a level of ambition comparable to Tracks A or B, pitch it. Email daniel.yarmoluk@gmail.com with a 1-paragraph description by **Week 8**.

SECTION 2

Best Practices & Insights from This Class

What we've learned so far that should guide your project:

1. Structure beats prompts.

The biggest insight of this course: the same model with structured context dramatically outperforms the same model with raw prompts. If your project just "uses ChatGPT," you're building a 2024 project. Structure your domain.

2. Show the delta.

Your presentation must include a before/after. What does AI output look like WITHOUT your system? What does it look like WITH it? The delta is the entire point.

3. Pick a domain you actually know.

The best projects come from people who understand the business logic. If you work in healthcare, build for healthcare. If you're in finance, build for finance. Domain expertise is the moat.

4. Build, don't theorize.

We threw out the slides in Week 1 for a reason. Your project should be a working system, not a PowerPoint about a hypothetical system.

5. Use AI to build the project.

This is meta, but important: use Claude Code, Cursor, or ChatGPT to help you build your project. Document how you used AI in the development process. That's part of the learning.

6. Think about who would pay for this.

Not required, but powerful: if your project solved a real problem for a real organization, who's the buyer? What's the ROI? This forces you to think about value, not just technology.

7. The audit trail matters.

Can you trace an AI output back to the structured data that produced it? Explainability isn't optional in enterprise AI. Build it in.

8. Start small, go deep.

50 entities done well beats 500 entities done sloppily. A 3-step agent that works reliably beats a 10-step agent that breaks. Quality over quantity.

SECTION 3

Recommended Tools & Resources

FOR KNOWLEDGE GRAPH PROJECTS (TRACK A)

- Neo4j (free Community Edition or AuraDB free tier)
- Python + NetworkX for simpler graphs
- Claude API or OpenAI API for LLM integration
- *The Intelligent Textbook*, Chapter 5: Custom GPTs, Agents & RAG
- graphifymd.com for inspiration on what structured intelligence looks like

FOR AGENT PROJECTS (TRACK B)

- Claude Code (free with Claude Pro subscription)
- Claude API / Anthropic SDK (Python or Node.js)
- OpenAI API + function calling
- MCP (Model Context Protocol) for tool integration
- *The Intelligent Textbook*, Chapter 6: LLM API Integration

FOR BOTH TRACKS

- GitHub for version control (required — your project must be in a repo)
- The Intelligent Textbook: yarmoluk.github.io/Digital-Transformation-with-AI-Spring-2026/
- Anthropic's education credits (if available)

SECTION 4

Timeline & Milestones

WEEK	MILESTONE	WHAT'S DUE
6 (Mar 19)	Project Proposal	1-page: domain, track choice, initial entity/agent list
8 (Apr 2)	Midterm Exam (Makeup Class)	Open-AI midterm — think with LLMs, apply what you've built
10 (Apr 16)	Check-in #2	Before/after demo ready, documentation started
12 (Apr 30)	Draft Presentation	Dry run in class — peer feedback
14 (May 14)	Final Delivery	Full presentation + GitHub repo + documentation

SECTION 5

Grading Rubric

CRITERIA	WEIGHT	WHAT I'M LOOKING FOR
Working System	30%	Does it run? Can you demo it live?
Domain Depth	20%	Did you pick a real domain and go deep? Not a toy example.
Before/After Delta	20%	Is the improvement with your system obvious and measurable?
Documentation	15%	Schema docs, README, process flow, how you used AI to build it
Presentation	15%	Clear, compelling 10-minute demo. Show the graph/agent, not just the output.

Grading philosophy: I care about ambition and execution, not polish. A rough but working system that solves a real problem will score higher than a polished deck about a theoretical one. Take risks. Build something you're proud of.

SECTION 6

What Great Projects Look Like

Examples of the kind of projects that would earn an A:

Legal Discovery Agent

Knowledge graph of case law relationships + an agent that traverses the graph to find relevant precedents.

BEFORE

Keyword search returns 200 irrelevant results.

AFTER

Graph traversal returns 12 highly relevant cases with reasoning paths.

Supply Chain Risk Graph

100+ entities (suppliers, materials, regions, regulations) with risk relationships. Feed it to Claude. Ask: "What happens if Taiwan has an earthquake?"

BEFORE

Generic answer about supply chain disruption.

AFTER

Specific cascade analysis through your supply chain.

Autonomous Code Review Agent

Agent that reads a PR, checks against your org's style guide and security rules, runs tests, and writes a review.

BEFORE

30-minute human review.

AFTER

2-minute agent review with human approval.

Healthcare Billing Optimizer

Knowledge graph of ICD codes, CPT codes, payer rules, and denial patterns. Agent suggests optimal coding combinations.

BEFORE

15% denial rate.

AFTER

5% denial rate with full audit trail.

Real Estate Investment Analyzer

Agent that scrapes listings, runs comps from a property graph, analyzes neighborhood trends, and produces investment memos.

BEFORE

4-hour manual analysis.

AFTER

10-minute automated analysis.

SEIS 666 — Digital Transformation 2.0 with Generative AI

Spring 2026 | University of St. Thomas

Instructor: Daniel Yarmoluk | daniel.yarmoluk@gmail.com

Office Hours: calendly.com/daniel-yarmoluk/30min

Intelligent Textbook: yarmoluk.github.io/Digital-Transformation-with-AI-Spring-2026/