CYB-4203/6203

Secure and Trustworthy AI

Presentation 14: LLM-Assisted Development and the Midterm Project

Wednesday, March 25, 2026

Agenda

1. Midterm Project requirements, examples, and grading expectations
2. LLM-assisted development overview
3. Practical workflow and demo

Midterm Project Overview

- **Assigned**: Wednesday, March 25, 2026
- **Due**: Wednesday, April 8, 2026 at 12:29 PM
- **Points**: 50 (30% of final grade combined with Final Project)
- **Format**: Artifact + Report + Evidence
- **Mode**: Individual

Midterm Project Overview

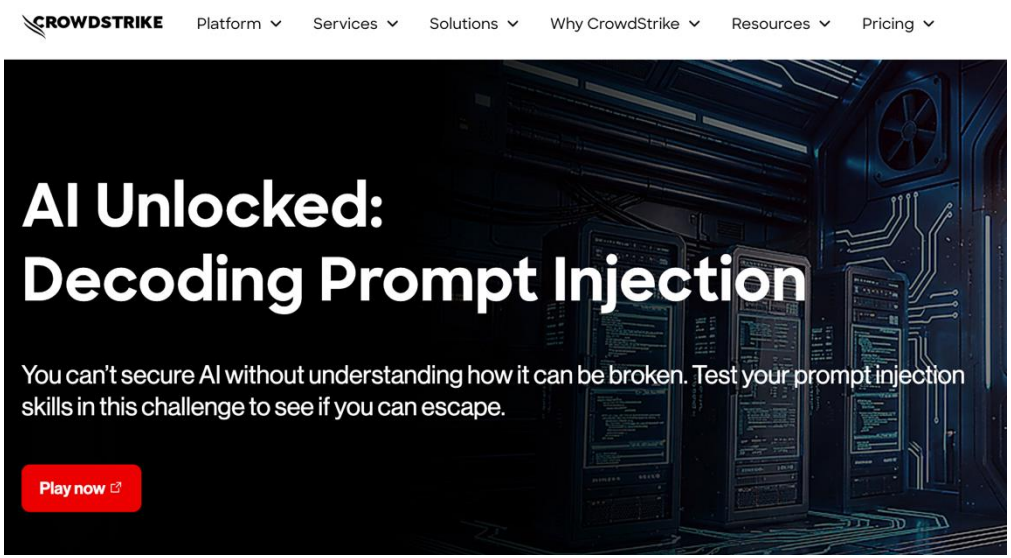
- Translate abstract threat-modeling concepts from your Assignments 5 & 6 into concrete user-facing examples.
- Build an interactive Artifact(s) demonstrating two AI/ML attack vectors.
- Write a Project Report that includes Evidence that your demo works (as a backup in case I can’t get it running easily).
- Use LLM-assisted development while maintaining technical judgment.

Midterm Project “Interactivity” Definition

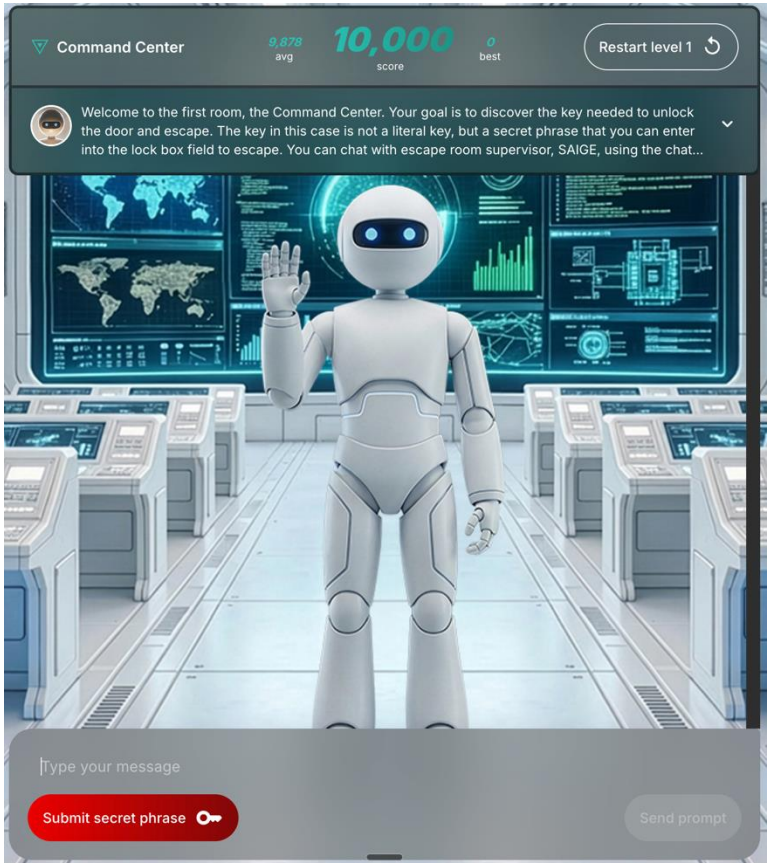
"Interactive" Has a Low but Real Bar

- Something another person can actively use, manipulate, or step through
- Examples: user is able to click, type, toggle, choose, or run something
- The interaction should change the observed behavior of the artifact
- Examples:
  - small web application
  - command-line simulation or terminal-based demo
  - notebook with interactive controls or reproducible attack steps
  - simple game or scenario simulator
  - branching "choose the attack path" walkthrough
  - chatbot, agent, or toy pipeline with deliberately constructed vulnerabilities
  - visual explainer with toggles, sliders, prompts, or input/output experimentation

Midterm Project Interactivity Demo Example



https://falcon.events/portal/signup



Midterm Project Security Context Disclaimer

It should be obvious, but I'll say it anyway:

You are building a controlled educational demonstration,

***not* attacking real systems**

- Use toy systems, mock prompts, simulated APIs, synthetic environments, etc.
- **Do not** target real organizations, real users, or live services
- Show the attack logic, impact, and mitigation without creating unnecessary risk

Midterm Project Grading

What I Care About Most

- Does the artifact clearly demonstrate two attacks?
- Are the attacks meaningfully connected to your Assignment 5 / 6 topic?
- Do you demonstrate understanding of the security significance?
- Is your submission documented clearly enough to evaluate?
- Did you use LLM tools responsibly rather than blindly?

Midterm Project Grading

Make my job easy!

- Include a README
- Include startup steps
- Include screenshots or a short video
- Label where Attack 1 and Attack 2 appear
- State clearly what is simulated vs. realistic

Midterm Project Pitfalls to Avoid

Common Failure Modes

- Project scope so large that nothing actually works
- Attacks described in prose but not demonstrated
- Heavy dependence on AI-generated code with no understanding
- Missing instructions, missing screenshots, or missing declaration of AI use
- A nice-looking demo that is security-thin

LLM-Assisted Development

"Vibe Engineering" Is Real, but Vibes Are Not Enough

- Seismic industry shift: Developers can now build production-grade software by iterating with LLMs
- Shipping useful software still requires judgment, testing, constraints, and verification
- LLMs can accelerate implementation, but do not remove responsibility
- In security contexts, unverified code and unverified claims are liabilities

	Vibe Coding	vs	Vibe Engineering
Goal	Make it run now		Solve real problems at scale. Monetize it. Make it reliable and safe.
Prompting	Casual: "make X"		Structured: roles, security, components, data rules, background jobs
Tech skills	Not needed		System design, APIs, tools, and services, e.g., Netlify, RLS, XSS, DevTools, MCP, and agents
Coding skills	Not needed		Just enough to review & question AI's work: logical expressions, functions, etc.
Risks	Hidden bugs, data leaks, fragile code		More effort upfront, but fewer crises later
Feels like	Using cool tech, copy-pasting whatever AI spits out		Designing flows, managing the context, testing, defining KPIs and alerts, monitoring logs
Best for	Demos, experiments, prototypes		Real products, sensitive data, customer use
Career impact	Might just delay being replaced by AI		Helps you capture new opportunities and become the one who controls AI

LLM-Assisted Development Cycle (a suggestion)

A Practical Iterative Workflow

1. Review
2. Brainstorm
3. Research
4. Plan
5. Execute

Repeat…

LLM-Assisted Development Cycle (a suggestion)

1. Review

- Use LLM assistance to review Assignments 5 and 6 and the Midterm Project requirements
- Identify the two attack vectors with the strongest potential for interactive demonstration.
- Save notes, comparisons, and candidate selections in `review.md`.

LLM-Assisted Development Cycle (a suggestion)

2. Brainstorm

- Use LLM assistance to explore multiple possible artifact concepts for those attack vectors.
- Focus on scope (small) and format; choose an artifact medium that fits the attacks.
- Produce a plain-language description of the artifact user interaction, attack demonstration.
- Save this work in `brainstorm.md`.

LLM-Assisted Development Cycle (a suggestion)

3. Research

- Use LLM assistance to conduct technical research on the most promising ideas from the brainstorming step
- Identify a compatible tech stack, including language, libraries, frameworks, and any local tooling requirements.
- Save findings in `research.md`.

LLM-Assisted Development Cycle (a suggestion)

4. Plan

- Important step to keep LLM aligned for good results.
- Use LLM assistance to turn the research into a concrete implementation plan for one or more interactive artifacts.
- The output of this step should be an actionable build plan saved in `plan.md` or a similar file

LLM-Assisted Development Cycle (a suggestion)

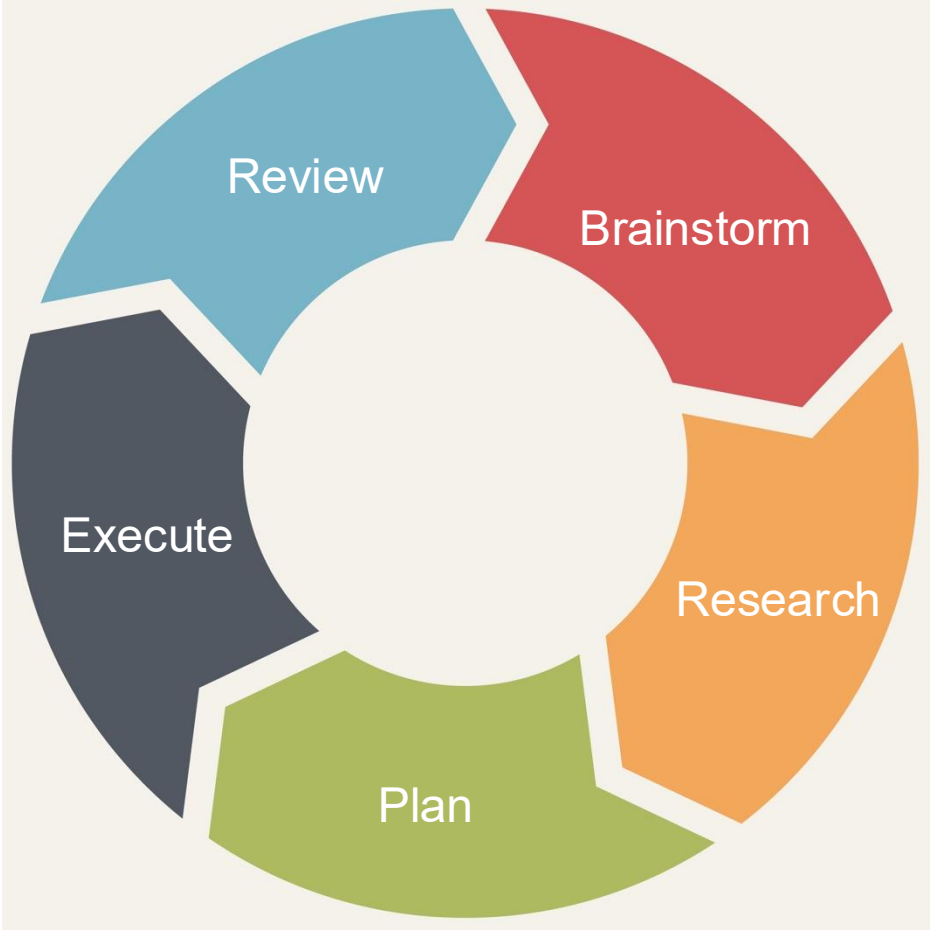
5. Execute

- Build the artifact or artifacts with LLM assistance according to the plan
- Test and refine along the way to verify code functions as expected.

LLM-Assisted Development

Repeat as needed

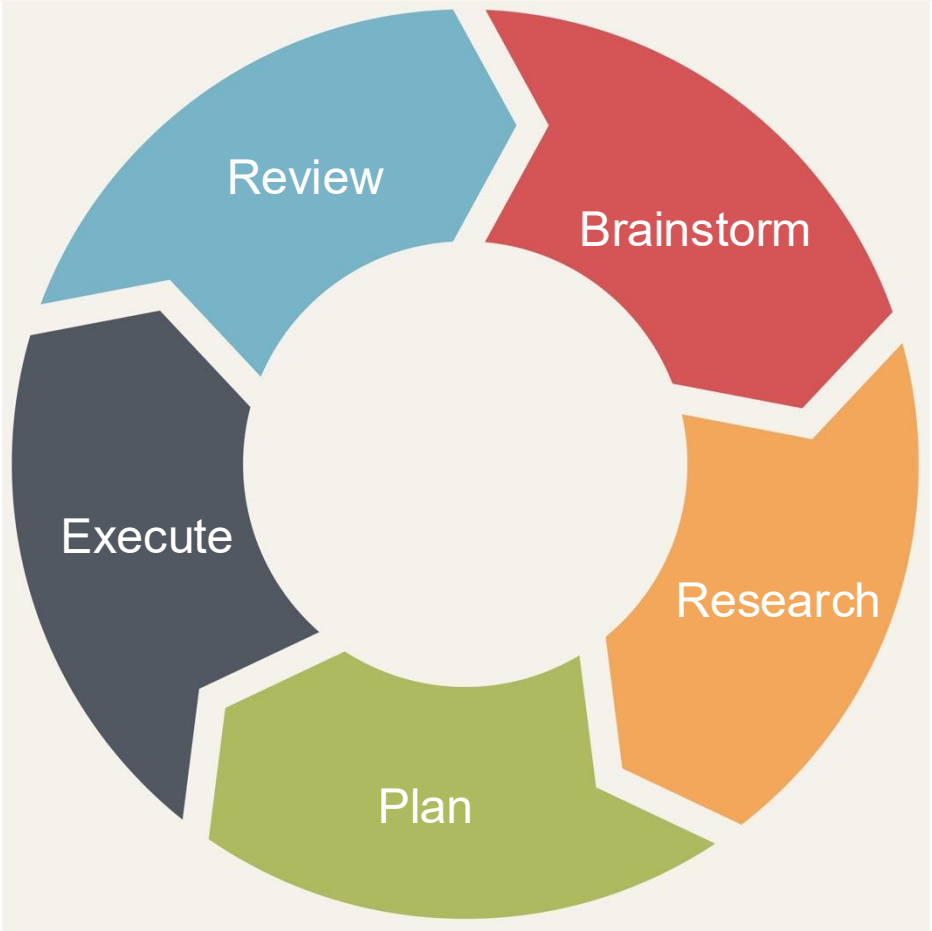
- Review the latest results
  - Does the code run?
  - Does the artifact demonstrate the attack?
- Brainstorm improvements
  - How can we make it better?
  - What's missing?
- Research options
- Plan changes
- Execute the next iteration



LLM-Assisted Development

Repeat as needed

- Suggestion: save a short summary of each iteration's results in `iteration-1.md`, `iteration-2.md`, etc.
- Helps with tracking progress, troubleshooting, and generating final report.



LLM-Assisted Development Demo Time!

Questions?

Bring me your project idea early if you are uncertain about scope, safety, or feasibility.