AI VOLS · EXECUTIVE BRIEFING

Comprehensive Briefing: Claude + Claude Code Basics

A shift from chat-based AI to agentic AI — ending with a portfolio site deployed live.

AI Vols · April 7, 2026 · University of Tennessee, Knoxville

Executive Summary

Meeting #3 marked the club's first hands-on encounter with agentic AI. Where ChatGPT answers questions, Claude Code takes actions — creating files, running code, and deploying websites. The meeting's demo produced a working portfolio site live on the internet before it ended.

Key takeaways:

- **Claude's differentiator is its context window** (200K–1M tokens), which makes long-form research and codebase work feasible in a single conversation.
- **MCP (Model Context Protocol)** is the universal standard behind ChatGPT Connectors, Claude integrations, and Gemini extensions — one protocol, many platforms.
- **Agent harnesses change the leverage ratio.** A single student with Claude Code can ship projects that previously required a small team.

The Claude Ecosystem

Claude Projects

- Mirrors ChatGPT Projects: a workspace with its own instructions and uploaded files.
- 200K–1M token context — large enough for a full textbook or codebase.
- Where: Claude sidebar → New Project.

Import Memory

- Carry over ChatGPT preferences and context without starting over.
- Flow: Settings → Profile → Import Memory. Claude returns a prompt to paste into ChatGPT, then the response comes back to Claude for import.

MCP — Model Context Protocol

- **One standard, every platform.** ChatGPT Connectors, Claude integrations, and Gemini extensions all speak MCP.
- **What it unlocks:** tools built for one assistant work across all of them.
- **Student relevance:** understanding MCP is sufficient — you don't need to build one.

Claude Code — The Agent Harness

- **Actions, not answers.** Claude Code creates files, runs code, browses the web, and deploys projects.
- **Three surfaces:** Desktop App (zero setup), VS Code extension, and Terminal CLI.
- **The shift:** regular Claude chats; Claude Code builds.

Live Demo — Portfolio Site in ~10 Minutes

- Prompt-driven build of a personal portfolio using Claude Code.
- Deployed live to Vercel during the session — publicly visible before the meeting ended.
- Iteration loop demonstrated: request changes, redeploy, refresh.

Key Concepts Reinforced

- **Tokens** — the unit of text the model reads (~¾ of a word each).
- **Context window** — how much the model can "see" in one turn.
- **Prompt** — the input you send.
- **Model** — the engine behind the assistant (GPT-4o, Claude Sonnet, Gemini).
- **Agent** — an AI that takes actions, not just chats.
- **Deploy** — putting a project on the internet.

Strategic Takeaways

- **The ceiling on solo output has risen sharply.** An agent harness + a thoughtful operator now produces what required a small team 18 months ago.
- **Start with the Desktop App.** It removes the terminal as a barrier for non-CS majors and preserves the full agent capability.
- **Ship something publicly within the first week.** The live deploy demonstrates the technique and creates portfolio evidence that matters more than any certificate.