

Claude Certified Architect – Foundations (CCAR-F)

Move beyond basic API calls and learn to architect resilient AI systems using Claude, Claude Code, Model Context Protocol (MCP), and agent SDKs. This practical, skills-aligned course equips software engineers with the architectural patterns needed to build and scale enterprise AI applications. You will configure developer environments using CLAUDE.md files, custom slash commands, and CI/CD pipelines, while mastering multi-agent orchestration, structured state management, error propagation, and context retention across complex agent workflows.

[CBT Nuggets course material](#) →

WEEK 1

Set Up Claude Code and Configure Projects

Installing Claude Code
Plan Mode and Project Context
File References and Project Memory
Challenge ☒

Define Claude Code and Coding Assistants

Introduction
Claude Code Memory Configuration
Claude Code as a Coding Assistant
How Claude Code Works with Tools
Goal-Based Claude Code Workflows
Challenge ☒

Apply Planning to Guide Code Changes

Introduction
Screenshots as Visual Evidence
Explore, Plan, Code, and Commit
Success Criteria and Session Controls
Challenge ☒

Create Custom Commands for Repeatable Workflows

Introduction
Custom Commands in Claude Code
Creating Custom Commands
Custom Command Arguments
Challenge ☒

Extend Claude Code with MCP Servers

Introduction

MCP Beyond Local Files

MCP Client and Server

Installing Playwright MCP

MCP Permission Boundaries

When to Use MCP vs Built-In vs CLI Tools

Challenge ☒

Implement Hooks for Safer Automation

Introduction

Why Hooks Exist

Where Hooks Fit

Hook Events and Configuration

Hook JSON Structure

The Hook Configuration — settings.json

Claude Code Hook Exit Codes

Challenge ☒

Apply Hooks to Block Sensitive Information

Introduction

Preparing the Hook Structure

Writing the Hook Script

Configuring and Testing the Hook

Challenge ☒

Apply Useful Hooks and Claude Code SDK Basics

Introduction

From One Hook to Workflow Control

Inspecting Hook Input

Choosing Hook Events

Claude Code Agent SDK

Challenge ☒

Explain Claude API Foundations

Introduction

From Claude Code to API Building Blocks

What the Application Handles

Basic Request Lifecycle

Setting Up the API Key

Making the First API Request

Challenge ☒

Send Structured Messages to Claude

Introduction

Why the API Is Stateless

Setting Up the Notebook

Single-Turn Requests

Multi-Turn Message History

Message Helper Functions

Challenge ☒

Manage Models and Controls

Introduction

Messages and Controls

Model Selection

Max Tokens Experiments

System Prompts and Response Controls

Challenge ☒

Stream Responses for Better Apps

Introduction

Why Streaming Improves User Experience

Message Helpers and Non-Streaming Baseline

Raw Streaming Events and Text Chunks

Challenge ☒

Control Outputs with Prompts and Stops

Introduction

Clear Prompts Lead to Better Results

Comparing Vague and Specific Prompts

Constraining Outputs with Allowed Labels

Prompt Specificity and Output Boundaries

Challenge ☒

Apply System Prompts and Temperature

Introduction

System Prompts and User Prompts

Creating a System Prompt

System Prompts as Operating Rules

Stable Rules and Temperature

Challenge ☒

Evaluate Prompts with Test Datasets

Introduction

From Prompt Guessing to Prompt Testing

Building a Test Dataset

Prompt Version One

Running the Evaluation Loop

Challenge ☒

Improve Graders with Model Feedback

Introduction

Code Graders and Model Graders

Evaluating Support Replies

Building a Model Grader

Evaluating with a Model Grader

Challenge ☒

Design Clear Prompts with Examples

Introduction

From Evaluation to Prompt Improvement

Testing a Vague Prompt

Clear and Direct Instructions

Using Examples in Prompts

Challenge ☒

Structure Prompts with XML Tags

Introduction

Testing an Unstructured Prompt

Adding XML Tags

Tagged Inputs and Reusable Templates

Challenge ☒

Explain Tool Use and Tool Schemas

Introduction

Why Claude Needs Tools

The Tool Use Loop

Building a Tool Function
Writing a Tool Schema
Challenge ☒

Handle Tool Calls and Tool Results

Introduction
Notebook Setup
Tool-Enabled API Requests
Routing Tool Requests
Completing the Tool Use Loop
Challenge ☒

Build Multi Turn Tool Conversations

Introduction
Notebook Setup
Message Helper Function
Multiple Tools in One Conversation
Building a Tool Router
Safe Tool Results
Challenge ☒

Route Requests Across Multiple Tools

Introduction
Notebook Setup
Routing and Message Helpers
Building Focused Tools
Inspecting the Tool Inventory
Completing the Routing Loop
Challenge ☒

Apply Built In Tools Safely

Introduction
Built-In Tools
Tool Types and Risk Levels
Classifying Request Risk
Deciding Whether a Tool Is Needed
Policy Checks and Tool Boundaries
Logging Tool Decisions
Challenge ☒

Compare Web Search and RAG

Introduction
Notebook Setup
Web Search vs RAG
Simulating Web Search
Simulating RAG
Retrieval Failure Modes
Challenge ☒

Understand Tokens, Embeddings, and Vector Spaces

Introduction
Tokens, Token IDs, and Embeddings

WEEK 2

Why Text Becomes Vectors
Visualizing Embedding Space
Challenge ☒

Explain RAG and Document Chunking

Introduction

Notebook Setup

RAG and Chunk Size

Overlap and Metadata

Keyword Retrieval and Grounded Answers

Challenge ☒

Build Embeddings and Vector Search

Introduction

Notebook Setup

Embeddings for Semantic Retrieval

Voyage AI Embedding Setup

Creating Embeddings and Comparing Vectors

Challenge ☒

Implement Vector Search with Voyage AI

Introduction

Comparing Keyword and Vector Results

Grounded Context for Claude

Metadata Filtering and Retrieval Evaluation

Challenge ☒

Implement Hybrid Retrieval Workflows

Introduction

Notebook Setup and Retrieval Data Model

Keyword Scoring and Exact Match Retrieval

Hybrid Retrieval Pipeline

Challenge ☒

Apply Image and PDF Inputs

Introduction

Choosing Image and PDF Workflows

Image Input Workflows

PDF Page Inputs and Representation Choices

Challenge ☒

Optimize Prompts with Caching

Introduction

Stable and Dynamic Prompt Parts

Estimating Cache Opportunity

Hashing and Verifying Cache Reuse

Challenge ☒

Use Code Execution and File Storage

Introduction

Reasoning, Computation and File Lifecycle

Building a File-Based Execution Request

File Metadata, Execution Results, and Artifacts

Challenge ☒

Explain MCP Clients and Servers

Introduction

MCP Hosts, Clients and Servers

Direct Integration Burden

MCP Primitives and Policy Boundaries

Discovery, Authorization and Tool Calls

Challenge ☒

Build MCP Tools Resources and Prompts

Introduction

Choosing MCP Primitive

Designing MCP Tools

Handling Errors, Resources, and Prompts

Challenge ☒

Test MCP Systems with the Inspector

Introduction

Testing the Server Boundary

Verifying MCP Tools

Automating Tests and Connecting Claude

Challenge ☒

Design Workflows and Agent Systems

Introduction

Choosing Workflows and Agents

Building Chained and Parallel Workflows

Building Evaluator Loops and Bounded Agents

Challenge ☒

Create Reusable Agent Skills

Introduction

Building Reusable Skill Packages

Building the Skill Registry

Reviewing the Skill Architecture

Challenge ☒

Build a Skill-Aware Agent

Introduction

Building Active Skill Context

Controlling Skill Resource Access

Running the Skill-Aware Agent Loop

Challenge ☒

Build an Agentic Multimedia Article System: Part 1

Introduction

Defining Contracts and Evidence Boundaries

Building Hybrid Retrieval and MCP Research Tools

Running Bounded Research and Grounded Drafting

Challenge ☒

Build an Agentic Multimedia Article System: Part 2

Introduction

Generating Assets and Evaluating the Package

Repairing Defects and Tracking the System

Packaging, Approval and Architecture Review

Challenge ☒