equip

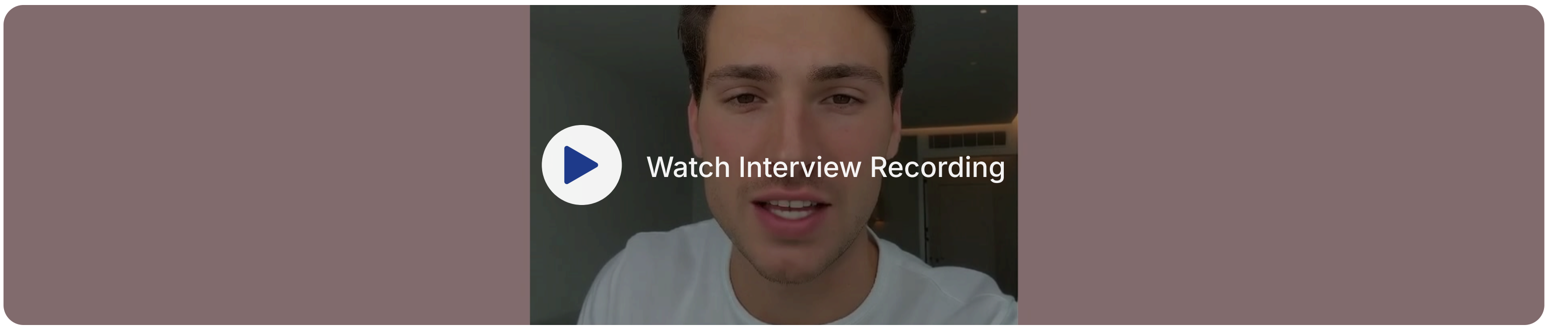
AI INTERVIEW REPORT

Alex Chen

Graded

alex.chen@techcorp.io

ROLE	DATE	TOTAL SCORE
AI-native Developer	10 Mar 2026	85.8%



RECOMMENDATION: STRONG HIRE

Alex demonstrated excellent depth across MCP tooling and AI agent architecture. Responses were specific, practical, and showed real hands-on experience. Recommend advancing to the final round.

STRENGTHS

+ Clear mental model of MCP architecture with real implementation experience
+ Strong grasp of agent patterns (ReAct) and when to apply them
+ Pragmatic engineering judgment — starts simple, adds complexity only when needed

AREAS FOR IMPROVEMENT

- Could expand on error handling strategies in MCP server implementations
- Did not discuss context window management or token optimization
- Agent evaluation and observability not addressed

TOPIC PERFORMANCE

MCP Tools

92%

+ USB-C analogy, built a real MCP server
- No mention of error handling

AI Agents

88%

+ Knows ReAct pattern, good judgment on when to use
- Didn't cover observability or eval

Context Mgmt

85%

+ Practical approach to large codebases
- No token optimization strategies

Code Review

78%

+ Aware of hallucination risks
- Vague on verification strategies

Prompting

82%

+ Understands system vs user prompts
- No few-shot examples given

INTERVIEW TRANSCRIPT

Bot
Hey there, how are you? I'm your AI interviewer. Are you ready for the interview?

Alex Chen
Hi there, I'm doing well. I'm ready for the interview.

Bot
Awesome. Let's start with MCP. What is the Model Context Protocol and why does it matter?

Alex Chen
MCP standardizes how AI models connect to external tools and data sources. Instead of building custom integrations for every tool, you get one universal protocol — think USB-C for AI. You have servers that expose tools and resources, and clients inside the AI app that connect to them.

Bot
Have you built or configured an MCP server yourself? What did it connect to?

Alex Chen
Yeah, I built one in Python that connected to our Postgres database. It let our Claude-based assistant look up customer records and pull usage stats on demand. I defined clear tool descriptions so the model knew when to call each query, and added row-limit guardrails so it couldn't accidentally pull an entire table.

Bot
Great, let's switch gears. What's the difference between using an AI model directly versus using it as an agent?

Alex Chen
A direct model call is one-shot — you send a prompt, you get a response, done. An agent loops. It reasons about what to do, executes a tool, observes the result, and repeats until the task is complete. That's the ReAct pattern — reasoning plus acting. The key difference is autonomy over multiple steps.

Bot
When would you choose an agent over a simpler approach?

Alex Chen
So if the task has multiple steps — like searching a codebase, reading files, making edits, then running tests — that's agent territory. But for a simple one-shot question, a direct call is way faster. I always start with the simplest approach and only reach for an agent when the task actually needs that autonomy.

Full transcript available in the non-demo report