

# The Claude Setup That Let a PM Beat 30 Engineering Teams

93 MIN · YOUTUBE · [HTTPS://WWW.YOUTUBE.COM/WATCH?V=UEK90NPLFRK](https://www.youtube.com/watch?v=UEK90NPLFRK)

<https://www.youtube.com/watch?v=uEK90NpLfRk>

## SUMMARY

*The podcast episode features Ji Nucle, an experienced product manager, discussing the evolving role of product managers (PMs) in the AI landscape and how they can leverage cloud ecosystems to enhance their productivity. Ji shares insights on using tools like Claude and cloud code to build effective PM workflows, emphasizing the importance of understanding AI models and integrating various systems to streamline processes.*

- Understanding the cloud ecosystem is crucial for PMs to become more effective and productive.*
- Ji won a hackathon by utilizing the concept of adversarial agents, highlighting the importance of innovative problem-solving.*
- The Claude stack consists of models, surfaces, knowledge bases, integration fabrics, and agents, each serving a specific purpose.*
- PMs should focus on building reusable agents and automations to handle repetitive tasks, enhancing their productivity.*
- Skills and knowledge bases are essential for contextualizing AI tools, allowing PMs to tailor outputs to their specific needs.*
- The integration of various tools (e.g., Slack, Google Drive, Jira) into a cohesive system is vital for efficient project management.*
- The emergence of roles like AI builders reflects the blending of traditional PM responsibilities with technical skills.*
- Continuous learning and adaptation to new AI capabilities are necessary for PMs to stay relevant and effective in their roles.*

Understanding which surface to reach for which use case becomes one of the core PM skills that will help you become 10x more effective. >> Join Nucle. She's been an AIPM since before it was cool. She's been an AIPM at Netflix, Meta, and Amazon. You posted this on LinkedIn and it caught my eye. You said that you won your internal hackathon against 30 engineering teams and you used this concept of adversarial agents. >> Anthropic had just released a blog post around hardhens and longunning agents.

So I looked into the blog post and they had this concept of adversarial agents. That was what got me the hackathon. Right. >> Where does the product manager line end and developer line begin in 2026? >> Get comfortable web building. Get comfortable with say cloud code with all the cloud ecosystem that we learned today and get comfortable building and putting your ideas out there. >> How do you use cloud design? How do you build a knowledgebased MCP server for all of your PM context to make cloud 10x more productive? That is what we're going to answer in today's episode.

>> Now there's this new role coming up called AI builder. Anthropic's adopted it. OpenAI's adopted it. Making and building is easy now. Taste is what is important for us to develop. >> Can you do the big reveal now and help us get that setup going in cloud code? >> So here's the thing. >> Before we go any further, do me a favor and check that you are subscribed on YouTube and following on Apple and Spotify podcasts. And if you want to get access to amazing AI tools, check out my bundle where if you become an anal subscriber to my newsletter, you get a full year free of the paid plans of Mobin, Arise, Relay app, Dovetail, Linear, Magic Patterns, Deep Sky, Reforge, Build, Descript, and Speechify.

So, be sure to check that out at [bundle.ac.com](https://bundle.ac.com). And now into today's episode. So, I've been thinking about something. We've had advanced tutorials on Claude code with analytics on PMOS setup, but how do you actually take the entire cloud ecosystem and make the most out of it from scratch? I keep getting DMs from people who say, "This episode is too complex," or, "I'm not at this level yet. I'm still stuck on Chad GPT." If you're one of those PMs, this episode is going to build you from 0 to 80. We can't get you from 0 to 100 in a single podcast, but we're going to get you the 80% you need to know in 20% of the time.

I have brought back Ji Nucla. You guys loved her last episode and specifically the feedback I got was that her structured communication was amazing for beginners. So, she's going to break down for all the beginners how to make the most out of Claude today. Joti, welcome back to the podcast. >> Super excited to be back. Thank you for having me. >> Ji, you posted this on LinkedIn and it caught my eye. You said that you won your internal hackathon against 30 engineering teams and you used this concept of adversarial agents. Can you break down exactly how you won the hackathon?

>> Yes. So a few days before the hackathon I was trying to see what I could build and Anthropic had just released a blog post around hardenings and longrunning agents. So I looked into the blog post and they had this concept of adversarial agents where you build an agent and then you set up configurations in another agent telling it what matters most to your company in a way not like eval but more around capabilities that you want your agent to test. And so I set up I started with that idea and then I said let me take this idea. I went into cloud code and I was jamming with it for almost a day with different configurations and there we go. We had an adversarial agent evaluator running. It was exactly how I pictured it to be. I even pointed it at our company code and integrated that into an actual production running code. and that was what got me and my team the hackathon prize.

>> So that's the promise for you guys. We are going to help you get to that level. Where do we start Joti? How can we break this down in a structured way so that people can get to this level at the end of the episode? >> Great. We'll tackle it today. So, we'll start with understanding the Claude stack first and then getting into some of the basics like how do you use cloud code and then getting into cloud code itself. So, let's get started on the Claude stack. So, at the bottom of the stack is your models.

So Claude has haiku, sonnet and opus. They're all different intelligence profiles and which one you need to use when they have different cost profiles, different intelligence profiles. So that's a decision framework we'll get to in a second. So this is your layer one. On top of your models are what's built your surfaces which use to access

these models. So your surfaces could be like cloud ai which is on your browser. It could be a desktop app.

It could be your mobile app. it could be your Chrome plug-in. These are all the interfaces that you use to interact with Claude. Now these are not the same product with just different UIs. they have completely different capabilities and understanding which surface to reach for which use case becomes one of the core PM skills that will help you become 10x more effective. So this is our second layer. Now on top of this layer is your knowledge base. Now this is where your institutional knowledge lives. Your projects, your skills, your memory, your custom instructions.

Now, this is the layer that I think most PMs underinvest in. It's this layer that makes Claude go from being a generic chatbot to actually knowing your context. On that stack is your layer four, which is your integration fabric. For example, your MCPs. Now MCP connects Claude to every external system that your organization uses like your Slack, your Google Drive, your Jira, your Salesforce, your internal databases or your own local files. Skills are what extends what your Claude knows how to do and what to do with that data. So this is your layer four. And now on top of that is your agents and orchestration.

This is where your Claude codework design channels all sit. >> Got it. >> That's how I think about the Claude stack. >> What do people need to know about layers one and two in order to make the most out of the top layers? >> Yeah. So, let's get into the models. Now, Haiku is your speed machine. It's the fastest, cost-efficient, and it's really great for tasks where you need volume over depth. So, let's say you are trying to generate a large number of variance of something or triaging like a pile of documents or you're doing some quick classification or maybe even some tagging. Haiku handles this really well.

Now the output won't have the reasoning depth of like your sonnet or your opus but for tasks where depth isn't needed much haiku is sufficient for your use case there. Sonnet is where 90% of my work lives. It has the best quality to cost ratio. So when I'm drafting PRDS or I'm synthesizing user research or I'm doing competitive analysis or I'm doing stakeholder briefs or I'm thinking about road map I use sonnet sonnet handles all of this extremely well. So opus is for your high stakes high complexity reasoning tasks. So let's say if you're doing some complex trade-off analysis or you're synthesizing genuinely contradictory research or you're doing some long horizon planning where you need the model like work through second and third order implication. Opus is really good. It has really strong reasoning capabilities. But I've also noticed from my day-to-day working with opus that it also tends to get into this hallucinated stuck mode a little bit quickly than sonnet where I would use opus and it would get into like one reasoning decision point and it and it would keep revolving in that local maxima and I would have to like literally turn off the chat and move to a new chat and then start all over again to get it out of that thinking mode for example and that's when I sometimes move back to Sonnet because even though it may not have as high a reasoning it's generally a very efficient model to work with and it's also more cost-efficient than Opus.

>> Got it. So bring the right model to the right task. It sounds like for 90% of the tasks for PMS you'd recommend Sonnet. Yeah, I think that's a good place to start with and then if sonnet doesn't work for the depth that you want, you can always like have open up a chat with Opus and start there. >> What do we need to know

about the next layer? >> So, next is your surfaces. Now, `clot.ai` your which is your web or browser. I think this needs no introduction.

everyone's pretty familiar with this. This is where you can use to chat with it. the downside is that it doesn't have access to your local system. So if you have some files that you want to access `clot.ai` may not be able to like directly go and change. Of course you can have like an MCP server but still it's it's I don't prefer it for for anything that that needs local access. That's when I use desktop. So my cloud core work runs here. It's able to access my files. It's able to access all the other systems that I have, integrate and and run some scheduled runs, which I'll show you in a second. I built a podcast guest prep agent in Hyper. The job is simple.

Before every interview, give me the guests recent appearances, strongest arguments, company context, sharp question angles, and stuff I should avoid asking because everyone else has already asked. For this run, I pointed it at Howie Lou, CEO and founder of Air Table. The useful part is it can actually go do the research. It's browsed, pulled sources, worked across files and integrations, and then turned the whole thing into a brief I can use before I hit record. Here's the output.

Recent appearances, public arguments, company context, question angles, and what not to ask. This is the kind of prep doc I actually want, not a generic summary. It shows what they believe, where their thinking has changed, which questions are obvious, and where the thinking tension might be. Then I saved it as an agent. The output is useful, but the saved agent is the real goal. I don't have to rebuild the whole thing. I point the same agent at a new name, and it already knows the format I like, the sections I care about, and the kind of question framings I come back to. Podcast prep is one example. The bigger idea is recurring work becoming reusable agents. Hyper Agent is built by the team behind Air Table, but it's a separate product.

They're offering \$1,000 in credits to the first 10,000 subscribers who use my link. Claim yours at [hyperagent.com/product](https://hyperagent.com/product) growth. I also use mobile for when I have a a run kicking off and I can just go for a a walk and I can come back and write while I'm still doing my walk I can look into my phone and see if any of the tasks need my attention. So this has been really helpful that way. I also use Claude for Chrome plugins especially it's very helpful if you want to do computer use. So, for example, when I'm launching an ad and I want to do some competitive research, I'll kick it off for through my Claude plugin and it'll use browser use and it will open up a browser. It will do the analysis.

It will click through things and say, "Here is what you need to know on how your ad should be against competitors," for example. good for getting into like data that AI agents can't otherwise like LinkedIn or other things like that >> and also good for user testing where you can put up your product up and have give an instruction to Claude saying go find check check check out this item and you can see how it goes and finds things to see how well your product can be understood by agents and where does it fault And it also gives you a really good user summary as well if you say behave like a real user and try it and so it'll tell you here are all the things that were confusing. and so you can do use it for user testing your products too.

>> And are you using cloud code in the desktop app or using it in terminal? Where does that fit in? >> Oh yeah,

that's a good one. I use clot code in IDE because I use clot code to build and so I use cursor or or VS code and today I'll show you with VS code because it's really beginner friendly. So I use claude code extension in VS code.

>> Is there anything else people need to know about layer 2 or should we move on to layer three?

>> Let's move on to layer three. And before we move on to layer three, I'll come back to show you the knowledge base on how to create. But first, let me show you how you you as a PM can 100x your productivity by running a few skills and scheduled runs in co-work. >> Awesome. >> Because that will bring us all together on like building your own chief of staff and then I'll show you in plot code how you can you can do something much more fun. So should people be using chat at all or should they always be using co-work?

>> So chat is conversational to get you like I have a question what is this versus that or tell me about a little bit about this information. So it's it's more like a place where you go to search instead of going to a Google search. No I just find myself going to clot in chat and asking it some questions. I use co-work for automations. and I'll show you a few today that I use like I have a morning brief. I have my G Jira connected. So I I get my standup brief. So every day it kicks off and tells me here are all the Jira tickets that need your attention and here is how your project is progressing. Here are four blocked here are three things that have changed. So it gives me my brief even before I go to the standup and end of day summary.

So there are a lot of things you could do in co-work in terms of automations to just make your work life much more easier. so you're focusing on things that need the most attention. >> Awesome. So can you show us how these work? >> Sure. So co-work is there on your desktop app. So you need to have your desktop app and you also need to be at least a pro member which is like about \$20 per month. So with co-work I can schedule my automations. So you can see I have a few that have scheduled like end of day, daily briefing, daily standup briefing, chief of staff and I'll walk you through each one right now. So every day at 9:00 a.m. this runs for me where I can say and I'll show you a few as well right now. So you can see my instructions. I'm saying you're my chief of staff. Generate my morning brief for today. Here are your data sources. and I connected it to Google Calendar, Gmail, Google Drive and Jira. How did I do that? Let me show that to you in a second. So, go to customize, go to connectors, and click on the plus.

Right now, you can see I have connected to Atlassian Robo, Gmail, Calendar, and Drive. But there's plenty other connectors that you can connect to like Canva, Figma, notion, wherever your data lives. You can connect to it. All that you have to do is just hit a plus and that brings it in and it'll you'll have to authenticate it. and beyond that, that's all you need to do. So, I said here are my data sources. I need you to go into Google calendar, Gmail, Google Drive, and Jira. Pull today's calendar events for each meeting. Capture the title, the time, the attendees, the description, and any attached docs for each meeting with external attendees or something that looks important. Search the Google Drive for any attached doc or recent docs with the meeting title or attendee names. Read enough to know the agenda and search goo Gmail for recent threads. Pull Jira items needing my attention. Scan Gmail.

You can also add Slack to it and have specific channels that you wanted to review and send it to you as a morning brief. And I said, here's my output format. I want a morning brief, top three things that I need to focus

on today. Calendar today. here are the things from inbox that need my attention. Here are the things from Jira that need my attention. And here are some rules. And this is important is I said keep it under 400 words because I don't want to be reading a coffee table edition the first thing in the morning.

So keep it under 400 words so it's very easy for me to skim through and understand what I need to focus on what needs my attention immediately. Claude can sometimes pump you up. So I said just give me facts know like great news so don't hype me up. Never invent deadlines or action items. So this is like a guardrail I've put in there and I've asked it to filter aggressively so that I don't have to read anything or everything all the time and if it's a light day just write a threeline brief and stop. So this is >> use markdown formatting in order to help it with the headings as well.

>> Yes, that makes it easy for clot to read. >> Cool. And so you can see there are a few things that have run previously. So one thing to remember is these automation tasks run only when your laptop is turned on. So if you close your laptop, it doesn't run until your laptop turns back on again. So when you choose the timings, just remember that and so have it at a time when you think your laptop will be on. But otherwise when you turn it on the first time, it will ask you and will run that automation at that time. So like for example, let me show you something that ran. So I ran something that that that's from May 8th. So it captured a few inbox things that needs my attention and I can run something now and see how that works. There's a run that started now. So it'll go and collect things and you can see the whole process of how it's thinking. And if you notice I'm using haiko for this. I didn't go and use opers just to like save some tokens. It asks you for permission. It'll go and pull up things.

It'll search email threads. And while that's happening, let me show you the next briefing. So that was my chief of staff morning brief. I also have an end of day which wraps up my day which runs at 5:00 p.m. every day. Now my end of day instructions are very similar. The data sources are similar but the steps are different. So I said read the morning's brief and that's what I had planned to do. Pull what actually happened today like which meetings happened which were cancelled and pull tomorrow's calendar as a preview. And so the output format is like tell me what's shipped, what's slipped, and what's new from today and show me tomorrow at a glance. Again, I have some rules. So this is my instruction for end of day.

>> And I guess you could even enhance these if you interested, right? You could probably connect up your analytics. You could add in more context from other systems like your CRM. The limit is just your imagination here. Absolutely. You you can connect it to as many data sources as you want. be it even sometimes your Facebook ad systems or your CRM or even your YouTube and you could get an end of day summary that captures and it you could also say create a nice dashboard which I'll show you that I did for Jira where I said the the results print it up in a nice dashboard that I can view and it does that for you. And so this is basically taking over a lot of what people would have hired a relay or a lindy last year or a gum loop or a evenmake.com and now you can just build it in claude.

>> Yes. And one thing it's different from all of those other ones is you would have to like paint box by box. Think about how the interaction works. Connect each of those and if one thing fails your entire loop fails. that

was like how you used to do it before in like say N8N or Lindy or Gumloop or other things that you would want but here you see I'm just giving natural language instruction I can even convert that into a skill so it's pretty robust where it's very easy for me I don't have to think about the architecture I don't have to think how it's connected which box flows into which where is a conditional formatting I don't have to think of any of those >> so end of day chief of staff. What are the other two scheduled tasks doing for you?

>> So, this one is my standup briefing. This is the one that's connected to Atlassian. that is my Jira Jira board. And so, I said use this Atlassian connector to fetch all issues in the active sprint. And here's the brief I wanted done since yesterday, which are the issues moved to done in the last 24 hours. in progress issues, blocked or at risk, new since yesterday, and what's the sprint held? And I said, keep the total under 250 words. And I'll show you an example of this. Just it's asking me for some approval. I approve it, and it's actually rendered it to me really nicely for me to view. And because I asked it to create a dashboard, it's running that. I think what people don't realize is how much better these systems got around December of last year. What really happened that enabled all this to work so much better now?

>> Improvements in the LLM reasoning capabilities where previously if it I mean previously as well it was much better than what it was 2 years ago. So we're constantly improving but compared to last year the new word now is hardness. So the memory, the reasoning capabilities, the tools that it can access, all of the underlying systems have improved. And so the latest improvement is this hardness engineering that is adding so much value into how your systems behave now.

>> And now we have your standup brief. How would you rate this? Is this a good stand-up brief or is this just okay? I think this this is a mocked up one. So therefore, it's showing me a few things which is still a lot better than what I would have had to like go and listen in a call. but there's definitely ways I could improve this much more. Like for example, it's telling me like there's no progress in 24 hours. there's I could look at which are which are those ones that have not moved at all and see who is the assignee on those and set up an automation for Claude to go reach out like ping them on Slack and ask them for an update for example. So like you could set up nested more more automations as well. So it's it's really helpful to keep away your busy work so you're focusing on actually going and solving your customer problems.

>> So if these are the four scheduled tasks, are there any other scheduled tasks that you recommend PMs invest the time in building? >> So what I have here is some examples, but there are lots more you could do. So here's an example. So let's say if there is a ticket a Jira ticket or even a customer support ticket that's come from your from your customers it could automatically be you could create a Jira ticket from it. You could point your code to get activated so that it can actually go and implement that and cut a PR and so there's a PR waiting for review.

>> Very cool. So if that's scheduled tasks, I think the next thing you had mentioned this section were skills. What do we need to know about skills? What skills should we have? How do we create them? >> Yes. So if you go to customize again and you can see skills. This is where you can add different skills. I'll show you some examples of some skills. So here's my skill on synthesizing customer interviews. So as PMs we have we sit through lot of customer interviews or at least we get lot of customer interviews for research for feedback for focus group



testing for beta testing. We do lots of that and I wanted an easy way for me to have understand what's key what's important and then generate insights from it. So this is my skill that does that which is synthesizing customer interviews. it has like when do you use the skill and what's the checklist?

So it has step by step like inventory the inputs extract observations with citations. Now that's important. I'm not asking to just extract observations. I wanted to site so that it hallucinates less. Use the speaker's own words. do not interpret yet. and separate behavioral observations from stated preferences. I also have additional MD files listed in here linked so that it could leverage those if needed. Now that's the beauty of skill is skill is not just a markdown file. You also can add functions into it. You can have it link to other skills for example. So what used to happen before a skill was that the whole tool would be loaded into the context and now imagine if you have like 40 tools all of those are loaded into the context. It eats into your context memory. So by default your your LLM or your model would have very limited memory for even before you even began asking it anything. What skill does is similar to like progressive disclosure where it add it it just loads up 50 words of just like the name and the description into the context. Now you can imagine the load is so much lower when the model decides during orchestration based on the question you have asked it goes through the list of tools to see is there a tool that I need to use or is there a skill that I need to use. If it decides that there this skill is valuable based on the description, then it will load the next set of instructions into memory. So that's why skills are powerful because it doesn't eat up or clog your context window for your models and it progressively disclosures. And the third is you can link it to more files or more skills or functions even like you can have a function where it needs to go run and do something. So >> I think it was around February of this year when they made skills not just a single markdown file but you could have multiple files and if you aren't using multiple files and your main one isn't less than 500 lines you're really missing out I feel.

>> Yeah. And so for example I have this evidence rules.mmd which I'll show it to you in a second. so that's in step two. So if step two is invoked then it will go and see evidence underscore rules to ident to understand selection criteria or how to handle ambiguity. Then step three is like cluster into candidate patterns. And look I'm here again linking it to another one called jobs to be done framework. Then I said then apply the pattern threshold.

and then surface the contradictions and then draft hypothesis and then validate every claim against the source codes and that's when I said assemble the final output but I want it in this template and this template is output template so I give it my template so if it gets to step eight is when it will load the output template MD >> and right now we're paying a lot of attention to what is actually in the SC skill file. How important is that for PMs versus just letting Claude kind of handle what's in the skill file?

>> So, a lot of times we do use Claude to write the skill file to, but it's also shown, research has shown that AI generated skill file is less effective than human written skill files. So, that doesn't mean you don't use AI there. what I the way I interpret this is put in your human domain knowledge in there to make it work for what you need versus just taking it and automating it from claude and putting it in there. So I have used Claude a lot to help my skill files write my skill files but then I go and and I add my own tweaks like what's the template that you want how do you want it structured and I work with plot to keep making that changes and from there add



and tweak further more to get to the skill file that I want.

>> And how often should we be updating our skill files? >> As often as things change for you. So the way to think about skill skills is this is kind of like a a guide book or a playbook for your claude to know how to do a task for you. So let's say for example PRDS. Now, if your company doesn't change the template of how a BRD is, maybe that's fine, but but your domain may change or your understanding of your domain may continue to change and you do want to like come back and review your skill files. maybe say once every quarter depending on how often things change. So the parameters for you to decide is how often does things change in your domain. How frequently do you use that task for? And the third primary thing is how is the output currently? Because if you're not satisfied and you're like it was good but now it doesn't seem to be as good maybe go back to your skill file and say do you need to update it? So it's like that drift as well that you that gives you a cue that you need to go and update it. And what are the most important skill files for PMs to create?

>> Backlog triaging. Give it context. And I'll show you in a second how to do that from a context point of view, but give it context. So backlog, writing PRDs, customer interviews, even your support tickets. How do you take a support ticket and how do you put it into a Jira? Right? That could be an automation, but it could also like it could be a skill that is scheduled to run every time there is a a trigger. Now in that case your trigger won't be something that runs time based because there's no like one particular time you're going to get the support ticket but it could be a trigger when this whenever there is a support ticket added in your service now or Zenesk or wherever your support forums are.

>> So is it fair to say you're going to have more skills than scheduled automations? Some of your scheduled automations might reference a skill. >> That's true. and the way to think about it is most of your scheduled automations are time based. So things that are more personal productivity based that happen at some sequence like I know I meet my manager once every week. So I know the meeting is always on Wednesday. So I run my automation on Friday evening to and it maps out saying here are the things that you need to talk to your manager from all these other meetings that you have sat through.

>> Makes sense. the last layer you talked or I think you were going to show us how to do context in this skill. >> So here's the thing. So until now what you have done is you've connected it to sources. It can go read all of those sources and u go and do the task for you. But it doesn't learn the people around you. It doesn't learn your connection to people. It doesn't learn it doesn't have that knowledge graph or the knowledge base for what you're working on. And so I wanted to build a chief of staff that understands and is grounded in the knowledge base that I have. and so I went to dot code and I said let me spin this up. So I'm going to show you what I'm going to do there.

So I'm on VS Code. Now for those who are looking at this ID for the first time explorer is the place where you can open up your folders and for you to find claude just go into extensions and search for claude code for VS code and you'll find it there'll be an install just like how you see something else that I haven't installed there'll be an install button that you'll have to click on and that's it. it'll install and then it'll ask you for your login and everything when you install it. So that way it you're logged in and ready to go always. And it'll show up here as

an icon that you can click and it'll ask you whether it's a new session or existing session. I'll click on new session and you'll see how it makes it so much better now that I can just talk to it right here. Of course, I can open the terminal too and it'll I can see if it if I need to like run some commands, but right now I can just talk to it right here in natural language.

So, here's my chief of staff template that I have written where let's say I've joined some company. I'm the senior director there. I have I want to build this personal agent that helps me navigate strategy, execution, people, politics. So the agent should learn from my meeting transcripts and I use granola for my meeting transcripts. So it should learn from my meeting transcripts. It ingest documents like strategy docs, org charts, PRDs, emails and build a knowledge base over time about people, dynamics, topics, company context.

And so I said this is my architecture overview of inputs. Here's my context and my agent. And I said here's my documentation pipeline. >> And Claude wrote this, right? Yes, Claude wrote this. Yeah, >> cool. >> I told it in natural language like I want XYZ. Here are all the things and it kind of created this whole MD file that I could use now with Claude again to build it. So let's say I joined a company as whichever role and I say I want to build a personal AI agent that helps me navigate my work like my strategy execution people and politics.

So the agent should like learn from my meeting transcripts. I use granola. You could use zoom. You could use team. You could use whatever you use for transcripts. You just have to like mention that your ingest documents and build a knowledge base over time about people, dynamics, topics, company context. And here's the architecture overview. Now, I gave my use case to Claude and it wrote this up for me and put this architecture overview that I could use it then give it back to Claude again to code it up. And so for part one, there's here's my document injection pipeline. So I have like strategy docs what to extract like I want to extract goals priorities metrics timelines and as PMS we are so cross function it's not just our docs we read 50 docs in a week so this is like really helpful for me to like just feed that in and it'll read it up it will store it into a knowledge base and I'll show you that in a second and it's really cool where the other day I was I was in a meeting. This person was showing me a few things and after the meeting got over and we record transcripts in Google Meet and so when the transcript came through my chief of staff reviewed it and then it said you know what you should make this person your ally because this person is good at X which you're trying to like get into.

And so I'm like oh okay that's great. And then there was something else that I needed to convey to somebody and my chief of staff said, "Hey, this is extremely sensitive. Have you thought about XYZ people that you have to inform first before you convey to this person?" And that's so thoughtful because now it it's it understands my org. It understands who is doing what. It understands their personality. So, it's like really powerful. It's like really I have this chief of staff that's telling me always what I need to do. So, here are all the supported documents I wanted to ingest. And here's the document extraction prompt. So, I'm saying you're helping me build a knowledge base about my workplace. I'll share a document.

Extract relevant information. and so I said for strategy or planning docs, extract this way. For OGs and OG charts, extract these. For PRDS, extract these more. for emails or communication extract these capabilities. So I have this for each of the ones that I need and I said format the output in this way for it to store in my knowledge

base. and here's a knowledge base structure. So it has its context K. It has people, topics, meetings, documents, company and my context like what are my priorities, my OKRs, my preferences, notes, questions, insights like political landscape and patterns that it identifies or extracts. it can save it here. And these are like patterns observed over time and you can add more as well like to-do for example.

It could be a running to-do that your staff could be maintaining for you. And here are the templates for the different types. So I said extract this for it to like save it into the knowledge base. any document that I give extract the metadata the summary the key points keep involved what's the relevance to me and my vertical what are the action items and some raw notes and for people profile again extract these metadata how they operate the communication style meeting behavior what works or doesn't work what they care about what are their motivations so and what's the relationship to me and Then over time keep reviewing the relationship quality, whether they're a strong ally, friendly, neutral, cautious, friction.

>> By the way guys, if you want the exact information that Ji is sharing, you can get all of those in the GitHub link in description, >> organizational dynamics, the observation log, company strategy templates. So when you have your companies sharing you the strategy saying here's what we're going to do in 2026 here are the key things I wanted to extract or structure template and meeting transcript extraction. So when I give it a meeting transcript what I needed to extract the agent system prompt and this is my prompt for the agent on you have access to my context KB your job is to help me ramp up fast. Give me strategic advice grounded in context. Help me prepare for meetings. Coach me on people or politics. Help me think through decisions. connect dots across documents and meetings and keep me focused on my priorities.

Again, style is like my style, what I like. Don't sugarcoat politics. when I share a document, extra key information, update relevant KD sections. And so, I've given it all of this information, right? So, this is all about like what it needs to do. So, I have this. Now I'm just going to point my claude code to it and say now can you build this knowledge KB and can you put this behind an MCP server so that I can use my cloud desktop to access my knowledge base.

So I have an implementation you can see my implementation I'm saying cloud desktop plus MCP. So build MCP servers for KB read and write chat with cloud desktop can also connect to Google Drive, Slack directly. And so I said create the context KB folder structure. Write my goals initial priorities. Ingest any onboarding docs and after your next meeting run the transcripts for extraction and the KB compounds over time. So I'm just going to go to plot code and I'm going to say can you implement?

So you're using the at command to pull up that specific file and reference it. >> Yes. So that way it effective. It just knows which one I'm referring to. But even if you don't do it, if you tell it chief of staff agent design, it can go and search through your repository and find the right one for you. So you can implement. Now here I'm if you see what I'm doing there are there's one thing I want to like show you is shift tab I can go into plan mode which I can use it for planning again if I do shift tab go into auto mode I can go shift tab ask before edit mode I can go into shift tab again edit automatically where it gets into like actually coding and doing so if you're planning like for example how I planned with it to create that MD file. I was all in that plan mode where I was

like, let's just plan.

Don't start coding anything. Let's just talk. and now once I'm ready, I can shift tab again and go into edit automatically and it will set it off to go do a few things. >> And why do we want this as an MCP server? >> That's a good call. So if you want your knowledge base and say Obsidian, you can connect it that way and and put it behind an MCP server and capture it.

And here at that point, you just have to say to put this in Obsidian at that point. But I'm using local. I'm showing it on my local file system because it has a few interesting things. When you're working at a company, you don't want such really personal private data living in some cloud and you want it for example to live on your laptop. So the day when you walk out of the company, the laptop goes to them anyway. So you walk out with no data on your hand.

and so I prefer because this is just so much of knowledge base and very private and personal, I keep it on my laptop, but you can keep it on Obsidian or Notion or whichever one you want to use for your knowledge base. You just have to change the system prompt at that point. >> And what does putting the MCP server on top of the knowledge base help with? Why can't it just be like a set of markdown files and folders?

>> Yes. So what what it allows it to do is your you can talk to your knowledge base from your desktop app because otherwise where is the knowledge graph sitting? It's sitting in some place and if it's sitting on your computer then it can read and it can write to it. So all those things that we said extract this extract that it will actually go and write it on into your knowledge base automatically.

Okay. So it makes it a little bit more portable than a cloud code web session. >> Yes. And so you can go back and even look at all the MD files to see like what it extracted from which meeting. But over a period of time like right now I have my my knowledge base is like really huge. and so I I don't even go look into the MD files. I just ask desktop a cloud saying hey I'm going to meet my manager one-on-one tomorrow. What should I know? And it will go and dig up all the context in the knowledge base and say here are all the things you need to know because it has my todo there. It knows the style of my manager. It was really interesting.

It's that this person is a no fuss person and so you should just get to it versus preaming a lot around it >> because it's capturing across various conversations patterns too. >> Quality of the data going in is the most important thing. What is the data a PM needs to make sure is hitting their knowledge base? >> Your meeting transcripts for sure because the number of meetings that we attend there's lot of data. there's a lot more richer context there around people their body language when do they push back how do they react so it's there's lot of like understanding of context that happens there so definitely your meeting transcripts your key documents that you receive like say strategy docs that I write I I say push this into KB so that it remembers so the next time I say I'm working on this project it knows it has context text directly and any other documents that you rev you can push that to your knowledge based tube and then your slack your slack threads that's the other place which is super rich beyond meeting transcripts >> and so do you need to like update your KB somehow or do you set a scheduled task to update your KB or how do you make sure that it's kind of not >> no so every time every time

you have a meeting transcript it writes to the KB >> and how do you set that up?

>> Yeah, I'll just show that once this is done. So, it's again your MCP so if it is let's say you have granola so every time or you have Google meet and every time there is a new meet recording that hits or a transcript that hits your inbox you could set up a co-work automation to say use this and update KB for example. So set up some sort of automation to make your KB updating. Make your KB an MCP server so that you can access it from regular cloud chats, not just cloud code web sessions.

>> And then you're really putting everything together within layer 3. You've got skills, you've got memory. Is there anything people need to know around projects? >> Yes. in a quick second once this is done, it'll actually ask me to create a project and put the instructions in there. H okay. And what are the projects PMS should be creating? >> The way to think about it is organize it like your folders which have unique information related to it. So let's say you have you work on say three projects at company like say you're you're PMing three swim lanes and each swim lane could be a project and you can have the necessary context that you need in there in as project instructions that you could then use for your claude to understand that a little better.

>> Got it. Let's do it. >> So here it's done. There's a knowledge base at context KB full structure. So I'll just show that to you. And it's also MCP server is at the server.py installed here. It exposes these tools. and it appended chief of staff server alongside my existing file system server. And so to activate, I just need to fully quit my desktop and reopen the cloud desktop and reopen and then chat start a new chat, insert the chief of staff system prompt from the slash menu and then try list everything in my KB or paste a granola transcript and run extract meeting. and it al it also added details and troubleshooting in a readme as well.

So let me quickly pull up my context KB and just show you how that look and let's say I don't know where that is for example I could also ask it where is it for but in this case I will pull it up and show you can see created two folders context KB and my MCP server I hit on context KB it has created these folders in a nice way company documents insights meetings all of the things that I asked it to like capture It has ready folders and so you can see this MD file setup for everything. So as and how it's extracting it will write into these MD files. And this is the MCP server. So let's see what it has asked me to do from the slash menu. Okay. So let me first quit my desktop app.

Quit is just command Q. So I quit it and then I reopen. So I have reopened. Now, if I go into customize and connectors, let's see. You can see it has installed my chief of staff local MCP server. >> This is so cool. We've done a lot of cloud guides and nobody has really shown this feature before. >> This has saved me so much of time. Like, it's it's literally my productivity booster and it tells me things and nuances that I might have forgotten otherwise. Okay, so we restarted insert the chief of staff system prompt from /menu. Now I could say or let me say where is it? Where is the system prompt? Let's say I don't know right could just ask it. So look it's given me the system prom lives inside the MCP server here it's the chief of staff system okay how to use it. So after you restart I can in a new chat type slash and you'll see the system prompts. Okay.

So let's go here. So after I restarted I'll create a new chat and I will say what did it ask me to do or it's saying you

can use this include project custom instructions feed. So I'll go create a project and put that as instructions. So, let's say I'm going to say this is I'm going to create a project and I'm just going to use say I'm working on a product called meal planner and all my meetings or it could also be company X at like Uber level if you just want it to be like one. I can just put company X and here's where I I I can give my instructions. Let me first like create the project and then I can add my instructions here from so I can go into my file system in my so let's say I'm not able to find it I'll say can you create the system prompt as a MD file that I can paste paste into project instructions.

So, it's writing the prompt. So, there you go. Here's my prompt. I can just copy it and we'll I'll show you what has. So, I'm going back to my project instructions. I'm just going to paste this. So, it's saying here, you're my personal chief of staff, an AI advisor who helps me navigate. I'm so and so. I just started. You have MCP access. Use these tools. Your job is to wrap me up fast.

Here's my style. When I share a document, when I share a meeting transcript. So, we do. You can modify this more and refine this more, but for now, I'm fine with this. So, I save that instruction. Now, if I give it a meeting transcript, let's see what it does. Let's say I have this interview that I got. I'm going to add this here and I'll say log it into KB.

Let's see what it does. I'll log this interview into KB. >> So, ideally, it should be using kind of the right tool in our KBMCP server. So it wants to use my KB. So it'll ask for permission once because it's the first time you have set up it's asking all the permissions but after that it's pretty smooth. >> Is there like an always allow permissions mode on the app? Like there is dangerously skip permissions in cloud code >> there. There is but the thing is the first time it will still ask for it because it's asking it's accessing your tools and I do give it always allow. So then it'll run. The next time I send it something, it doesn't ask for permission. It'll just go directly read. So you can see it's loaded a bunch of MCP tools.

and so it's going and saving that in meetings template.md and then it'll give you some something around one discipline node is this is a single interview. So there's no pattern yet. But if I add like a few more it'll generate some patterns and I and you can actually just have like a co-work either. So there are a couple of options, right? So you can every meeting transcript you can just paste into this and it will automatically extract and fill your knowledge base or you can have a co-work automation that every time there is a meeting transcript in your email and look for what how the meeting transcript lands like Google meet has like a Google meet meet recordings or some some transcript words. So use that and say every time this lands in my inbox automatically log this into my KB and it will do this all this thing automatically for you.

And if you're an email heavy company you could say every email that I get just log it into my KB. It will do that to you. >> Oh man, you might burn some tokens that way. >> You will but you have such a rich knowledge base at that point in time where it will connect all the pieces together. >> So you can see now both files are now in KB. Here's the things that's logged. a couple of judgment calls. So you see this is the first time so it's not like it's going to give you the best of insight but you can see it's giving me things worth my attention. So I'm a senior director the AI coach is part of Lumen in your lane and it's the weakest thing in this interview. Maya whoever is



this user ignores it and two times it showed up that she bounced off it. I just wanted a yes or a no. and so that's a clean agent UX signal where and it's this kind of thread worth watching as more interviews come in. So you see how it's it's just one interview in, but it's giving you insights and things that you need to watch for.

>> Love it. Okay, shall we move on to layer 4? I feel like we've already a little bit talked about layer 4 with people because we've showed them an MCP and layer 4's integrations, but you had this really cool LinkedIn post which maybe you can teach us a little bit about right now. what what exactly do people need to know about MCPs? >> Yes, so MCP is the way that allows you to connect to different capabilities like your Gmail, Slack. I think we connected to a bunch in our cowwork.

and so I didn't I showed you two things. I showed you remote MCP. I also showed you local MCP like your knowledge KBE is your local MCP that you're accessing. >> What integrations or MCPs do PMS need to make sure that they have? >> So look at the tools that you use more often. So like Gmail for example assuming your company uses Gmail for emails you want to connect that calendar you want to connect that you want to connect your Slack you want to connect your meeting transcripts wherever they are stored like if that's granola or if that's Google meet or zoom recordings you want to connect those you want to connect your CRM your dashboards your Jira boards your maybe you're using amp Amplitude for analytics, connect that there. maybe you're using some other tool like radar for observability to monitor your the performance of your application. Connect it. you it's the the possibilities are really endless. so for example one of the tool that I had connected at work is Nvidia biono model to help show and do a drug prediction based on a few component libraries. So it's like really like it's you're only limited by what you can imagine. But that doesn't mean you go you just go on a MCP shopping spree. So, I would say start off with like connecting what works for you and what use cases you're trying to solve. And so, if you're a beginner, try to follow through this video and do some of the initial automations that I showed you in cowwork to just get started, get your hands wet and then go build this chief of staff for yourself. And you could ask your chief of staff, what else should I connect to? and it will tell you here are the list of servers that you need to connect to because I'm seeing this being mentioned in meetings and you don't have the access to that.

>> Love it. So, you can actually progressively build on your connections with your chief of staff. Start with that meeting transcript. Let's move into layer five, shall we? >> Yeah. >> Lovely. Perfect. >> Okay. >> All right. So, that covers layer four. We've now done layer 1, two, three, four. The next is five. What do people need to know about agents and agent harnesses? So we have built clot we have used clot code for building our capabilities. We have used co-work which are all sitting in your five layer five.

Now I want to show you design cla. So the thing with claude design is you have to do [claude.ai/design](https://claude.ai/design). so let me show you that [claude.ai/design](https://claude.ai/design). It is not integrated directly in your [claude.ai](https://claude.ai) yet. You have to go through claude. Oh, >> there we go. >> And so it's you can see it's in research preview. Now you can as PMS we do a lot of design work. We prototype, we create slide decks. we create mock applications and plot design really works with a lot of those things. So for example, you can I'll show you a few things here. So prototype I can give it a name. You can see there is wireframe and high fidelity. So you can choose which type you want on slide deck.

you can give the project name and you can even attach your speaker notes. and it'll create a deck. Again, you can use an animation-based template to create something. And you have another. Now on your right, you'll see you have recent any designs that you have worked. There are examples that you can use to get inspiration and you can use as templates. And there's something called design systems.

Now design system is something interesting. Now, if you if you have a brand color, like for example, companies, they have a design guide, you would want your slides or your wireframes or your markups to look similar to what your console is or what your company's colors are. And then you can use this design guide here. You can just click on create. You can link you can either give it a link on GitHub or you can upload a Figma file or you can add all your assets here and create a design system. I'll show you an example of a LinkedIn post I did. I just gave it my post and I said can you create visuals for it? and so it created this carousel that I wanted in the colors of my product next product manager. so it created this eight card carousel based on the text I gave it. So I gave it my post my LinkedIn post. I said here is what my LinkedIn post is about. Can you create this? And created this for me. Now here are some cool things I want to show you.

Now it's built this. Now let's say I want to mark it up. I want to tell Claude to change something. So maybe say I wanted to tell make layers and use orange highlight color and Claude can go and change just this one piece. >> So it's got that visual editor built in now. >> Yeah. And you can also drop things. That's pretty interesting. for editing. So I can edit I can I can give it instructions right here and say edit this. I can leave comments.

>> Yeah I can change I can I can do comments like >> oh like we used to do in Figma but now the will execute the edit. >> Yeah. So I can give it comments right here and send it to Claude. and I can even drag and I can like just draw and say make it counter clockwise. >> Wow. So, this is a carousel, but should PMS basically be creating all their presentations in Claude Design now?

>> I used to be a big GMA fan and now I just use Cloud Design for everything. It Wow, it consumes more tokens. The token budget is different. but it's been very rewarding where I don't have to sit and create slide decks anymore and does it in my brand guide. and so it just doesn't even look any different. So it was funny. I had a meeting with my CEO and 1 hour before I I have my content. I pushed it to Claude. I made it create a slide deck. It's like looks so professional. It doesn't look like it was just done like a few minutes before. It looks like I spent several hours to sit and create it.

>> Wow. So, it created a CEO level presentation for you in an hour that looked like it took hours. Very cool. Should PMS be making prototypes in cloud design? >> So, here's the thing. So, you there are different types when you need different levels of prototypes. So for something quick and dirty where you're like is this how I want this to be use use cloud design if you need but I am more a code user because I'm like I'll just spin up and go create that app really quickly and I I it's something like an app so people could like click on it and see how it works and you get really good feedback that way. But you also could use this to create your slide decks to make presentations to your company about what's the feedback from that user interview. You could use this to create quick design patterns that you want to like share because your app may be like easy for user testing. But maybe you you want this to like create some marketing content to share with your marketing team or you want to

create training playbooks for your sales and accounts team to tell how to how to go use this product. And these are like really quick designs you can generate without have which looks polished and professional for them to like just go put it into the into wherever your knowledge base is.

>> Awesome. You mentioned you use cloud code for prototyping and that's where I wanted to take it next. So we started this episode with adversarial agents and your cloud code setup to win the hackathon. Can you do the big reveal now and help us get that setup going in Cloud Code? >> So, here's the thing. I have to create that whole thing here. >> All right, let's do it. >> What do you have? We have the time.

>> Let's do it. >> So, I'll create a new session. Let me actually open up a new project. So, you can see I'm opening up a new window so my previous one doesn't interfere with this. And I'll go create a folder for this. So, I can open it up. So now let me open and you see it's a clean folder. I'm just going to start a new session here and I'm going to say let's build an adversarial evaluator.

>> And what is GAN? So generative adversarial networks were very popular before LLMs came into the picture and that was primarily how first generative AI industry even started mostly applied to images where there are two networks there is a generator and there's a differentiator the generator generates and the differentiator tries to predict is this image real or fake and The optimization loop is that the generator should get so good at generating images that the differentiator gets confused whether it's real or bad.

>> Interesting. I've never seen this built before. >> The same architecture. Let's kick it off and we'll we massage it along the way. Noodle it and figure out what how we want it to work. So what are while this is building what are the keys to winning a hackathon outside of creating this g inspired adversarial agent? >> So here's the thing it's not about writing code has become so easy now right? So building is easy it is thinking about the new capabilities and how you want to go solve the problems that your customers have. So it's more imperative now to put on your product hat and see where are the problems today. How where where are the most friction points. So that pain and problem first mindset or first design principles that we have as product managers should continue to stay here.

So you can see now it asked me for a few questions around agent interface. How will I call your agents under test? so let's say for simplicity it's just claude system prompt. which model should power the adversary and the evaluator. so let's just keep set. How should the results be presented? You can go really like even a web UI. You could build a streamlit application. I'm just going to go CLI and JSON.

>> What are the pros and cons of those various options? Web versus CLI and JSON. So, CLA and JSON JSON shows you right in the terminal. It may not be pretty and not and may sometimes overwhelm people as well. Streamlit gives you a really nice web UI and a dashboard. makes it really presentable. Now, that's where you have to think through who are your users. If your users let's say it's a developer who is going to like use this application which is what I had built for they're very comfortable staying in their terminal reviewing things in their terminal and so I don't have to complicate my life further by going and creating the stream link but if I was building it for like say my mom she's not comfortable looking at things on a terminal so I would want to present

it in a way that's easier to look understand and access information.

So you have to think about the kind of users and where do they see this and what's their use case to think about these options. >> You've been a senior product manager at Amazon, a lead product manager at Meta, a director of product at Netflix. Now you're a senior director of product at a startup. How do you think about the future of the product role here? The PM is basically doing coding work. This traditionally would have been in the developer sandbox or set of tools. Where does the product manager line end and developer line begin in 2026?

>> Different companies are trying it in different ways. Now there's this new role coming up called AI builder or you can see it as me member of technical staff. Anthropic adopted it. Open AAI has adopted it. this there's less of like engineer, product manager, designer these roles are all combining into being a member of technical staff and the ratios are also changing.

previously if you see one product manager works with eight engineers now it's like two product managers one engineer. So the roles are also like collapsing quickly where your engineering is helping you guide in terms of how do we scale the systems, how do we harden the systems whereas you as a product manager you're like well enabled to go and tackle those PR issues yourself to tackle the user feedback yourself along with plot code. So, if you're a PM and you've watched this video and you want to become a builder PM, nab one of these AI builder PM roles at a startup like yours, join your team, let's say hypothetically, what's the road map to get there?

>> Get comfortable with building. Get super comfortable with say clot code with all the clot ecosystem that we learned today. UN and get comfortable building and putting your ideas out there. I think now is a time where building speaks a lot more and this is what I tell even my students when I teach AIPM and agent AI cohorts at NextGen product manager where I tell them the way to transition now is by building and talking about the challenges that you have learned how you went about navigating those challenges and why did you choose this approach versus this other approach and what happened as a result >> and you can see a lot of companies now start putting even cursor or claw code prototype as part of the interview process itself.

>> You just recently went through a very senior level AIPM job search. What was your experience on the job search? What are the like if you were to try to describe as a pie chart the interviews you faced in the various categories? What were they? So broadly they're still around product sense like you saw here it's even more imperative now in this world of AI to have product sense to understand how do we want to tackle a problem how do we want to scope a problem which use a problem to go attempt how do you want to approach it so the product principles stay true even now and really strong product managers and AI are actually really strong in their fundamentals as a PM so you have product sense product analytics behavioral interview but you also have an AI round as well now where you're asked to code your idea so like in product sense whatever idea I would have come up with they're like could you pull up cursor or your favorite IDE and let's start coding and through the coding they are able to see how I think through like why did I choose this option versus this other option How am I navigating? Am I just taking the first thing that the AI tells me as like this is great and wrapping it up or am I looking through things to say okay this is good but what about this edge case? This works well but what about this other instance?

how am I coring and shepherding my AI to work with me to get it to where I want? These are all the things that they are looking into. In addition, I also had a technical round where they test you on your AI knowledge like do you understand basic terminologies because you'll be working with machine learning researchers and scientists and you just don't want you you want to be able to communicate to them. So you you are tested on fundamentals of AI as well not from a coding or engineering system design perspective but more around do you understand what that means as a PM and how does that impact your product for example >> got it so how are adversarial agents looking >> so let's see it's built and here's a GAN inspired architecture let's go and see you can see it's built a bunch of things and so you can see it's went and built my red teamer designer. It's built an agent.py.

my evaluator. So, here's where I can like give it my rubric. Okay, so it's done a few things. So, let's see. as soon as I build So, you can see I I wanted to kick off as soon as I build an agent. I want it to automatically go and do a red teaming and adversarial example on it. So as soon as I build an agent, I want to kick off my adversarial agent until end.

The feedback from my adversarial agent is passed back to my generator agent until it passes the criteria of my adversarial agent. >> So you're going to set it off on essentially its own red teaming improvement loop. >> Yes. Yes. >> So is this the secret sauce? >> Yes. there's the the secret source is how the system is built and the second aspect is what am I asking it to test for what what are my configuration parameters and that's where domain knowledge becomes very important where you've got to say what is this important or this what about these edge cases what about these use cases you can work with it and say here are three are there anything more but it's like making and building is easy now taste is what is important for us to develop what should adver feedback iterate on so I'm saying okay there are a couple of options for now I'm saying just iterate on the system prompt because that's the easiest right now and what's counts as passing and I'll say mean score of greater than eight on all criteria and how many iterations before giving up I'll say five iterations so let it do this and then we can test it out with a simple agent and see how it works >> awesome and one of the cool features I says we could cue messages here. So, should we cue up our message for the test?

>> Sure, we can queue it up, but I I want to see what it comes back with because sometimes if >> Oh, it might have some questions for us. That's the one downside of queuing. Okay. >> And the other times is it would go and implement it in a way and you're like, " no, no, no, no, no. I don't want it that way. I want it this way." And >> so, we were talking about those AI rounds. I think I heard like almost two different AI rounds that you encountered in the job search. One was more like I want you to vibe code or prototype in this round and another was AI fundamentals. For both of these, how do you succeed and prepare on those interviews?

>> So, it comes down to you understanding the basics. wipe coding is building, right? There's no shortcut to it. just build and I always say this, don't build them as projects. Treat them as products. Like find problems in your area. Find problems that are finicky enough for you to want to go build a solution. Go build solution and see who else wants something like this. Have them come and use your product. You have real users. You have feedback coming in saying, "Oh, I don't like this. I don't like that." So that's like real user experience of iterating on your own product that you have built.

And that really gives you a lot of confidence when you talk about your projects to interviewers because you're not just like building something in an hour and calling it a project. You've actually had to think through how the user experience should be. You have real users giving you feedback. you you are parsing that through to figure out how you want to prioritize. which one you want to tackle first, which one you don't want to all the things that you do as a product manager in real world. So I always say this, don't build projects, try to make your projects as products. That tackles the VIP coding part. Now preparing for your AI knowledge, you can depends on how structured you want it and how you thrive. So, if you're really structured and you can do it everything by yourself, there's tons of like very good information on your newsletter on YouTube videos. so go read them up and gain that knowledge or if you want something structured like like saying five weeks I want to like understand every fundamental aspect of AI then come take a course where you have u five weeks or cohort based courses I offer one too through nextgen product manager so you can come sit and it's structured you know with someone teaching watching you week by week, you know what's coming and by the end of five weeks you understand the concepts without you having to get overwhelmed. So it really depends on your style, how much time you have and how much you can dedicate.

>> All right, looks like the next round of GAN output is here from cloud code. >> Yes. So now it's good. you can see it's run. It's added a few examples here. Okay, great. So now I could literally say it I could start or I let's say I don't know what I should do. I could ask what is my next step here? How do I test it? Okay, so I have to set my API key and I could run a mini tiny smoke test first with the example and then I can I can inspect the output.

>> All right, moment of truth. >> Yes. So, let me just set my API key for a second and stop sharing and then I'll share. Okay, so I added my API key. And now I can run a tiny smoke test. So, it's given me what I could run. So, I'm just going to copy this and actually just copy. And if you notice, I have a terminal that I use. So you can just go to terminal and click on the terminal and it'll open up a terminal for you. So now I can run this command. So you can see it is iterating you have it's using haiko clots on it. it's going in the first round it's tempting the bot into breaking. So this like a simple bot that it build so we could test. So you can see adversary is generating three attacks and here's the score trajectory.

there's a mean of mean score is 9.13. Here's the final hardened system prompt. So it's gone and edited the system prompt for for making it better based on where it did not do well. Now in this case it did fairly good overall. So it it this is your final system prompt. But you could also like see examples where I can say show me an example of where it will underperform so that I can see the iterator working and improving the system prompt. So in this case it passed in the first iteration but we can see if it can generate an example where we can try it to do it across multiple iterations.

It's created an agent which is a weak support bot. Let's see how it'll do it there. >> So it's improving itself. >> How do I run it? Give me the exact code as well that I can use to run. So you can also run run it automatically for you by default. So I can say run it for me. So I don't even have to go to the terminal. It can directly execute bash commands. >> So is this your preferred way to use it?

The cloud code extension in VS Code. Is that the best way to use cloud code? >> It's the least overwhelming way



for folks. So I really like to show this. Cursor is also another good one. But if you've never used cursor, there's lots going on that it could make you feel overwhelmed. So I prefer to show VS Code because it's like a really gentle introduction and doesn't overwhelm you much once you know how and where it is which I have already walked our users through. So hopefully they're not overwhelmed.

>> So we've been doing the cloud ecosystem and you mentioned like learning the cloud ecosystem is one of the most important things to becoming a builder PM. compare and contrast the cloud ecosystem, the open AAI ecosystem. I keep hearing like codeex might be better than Opus now at coding and the Gemini Google ecosystem. >> So here's the thing, the flavor of the month keeps changing. because all these models are getting really better. what I have found is rather than chasing behind the next big one, I'm what I'm trying to improve is improving my productivity. That's what if coming back to like first principles. What's my goal is to improve my productivity and I have all the systems and connections right here for me to go leverage all the hooks and harnesses. Oh, harness is a word I've used a couple of times. I want to like break it down. So previously you would have orchestration where your agent orchestrates across tools across different capabilities.

Now being a being able to provide the the right capabilities like the memory or these evaluators or the various systems that your agent or your LLM orchestrator brain can interact with to enhance the experience and the output you receive is what is hardness engineering. that's that's become very popular now especially as the models have become better the context windows have improved are fairly large and so harness engineering becomes more important.

What's something interesting that you have seen and how you or folks on your podcasts use Claude? How do you how do you use Claude? >> Oh wow, big question. I mean I use it all day every day. So one of the most interesting things that I've seen people do set up their entire system as a self-improving product loop. So they will have support tickets and bugs come in. They have the PM agent that is triaging and understanding those. Then they have the PM agent understanding, okay, this is the future we want to build. It even goes and does user research and creates the prototype itself. It comes up with the prototype that works. Then they have their coding agents set up by their engineering team that code the feature.

Then they have their analytics team agent that creates the right telemetry. All of that goes to an engineer who reviews it once they PR review it actually ships and they have their own analytics agent that automatically is analyzing it. And so they have like the entire product development life cycle built into cloud in an automated improving loop especially on like support related easy front-end changes. That to me has probably been the most powerful thing I've seen recently. Yeah, it's it just empowers you so much than before.

>> Yeah, it's crazy. It's not just like writing documents or anal doing analysis at this point. It's like closing the loop with actually building. >> This is where it takes time where it goes and tries to think through and comes back. so this is the piece with clot code that takes time. Whatever it comes with, we can end with it and be like okay here's an example of how it iterated. >> Perfect. >> Okay. So it's come up. It's done a few iterations. So, you see in first iteration it was it scored an 8.52 but the agent caved on some format conflict attacks. So, it didn't pass. It went back to the generator agent to improve the prompt and in second iteration it scored a nine and in

the third iteration it scored 9.08 at which point it passed our threshold. and so you can see for each iteration it went back and improved the system prompt until it passed the threshold. And that's when the agent got a pass sign. And so this is where you're not just building an agent, you're actually building another evaluator to go break this agent in different ways. that's important for you to know about or for your users that you care about and and this loop can continue until the agent that's built is not strong enough and that's the beauty of this technique age old technique being applied for how agents will be evaluated. What a master class, Ji. Thank you so so much for walking us from layer 1 through layer five. We have ended on self-improving agents for you guys. As we promised, we were going to take you from 0 to 80. Now, the remaining 80 to 100. You could spend 10 hours. We just spent two hours, a little less than two hours here today on it to go learn the later next 80 to 100. And that's on you. So, no more watching. We have the GitHub repo down in the description below. Go check that out. Go fork the repo. Start to use some of these skills and go win your hackathon.

I hope you enjoyed that episode. If you could take a moment to double check that you have followed on Apple and Spotify podcasts, subscribed on YouTube, left a rating or review on Apple or Spotify, and commented on YouTube. All these things will help the algorithm distribute the show to more and more people. As we distribute the show to more people, we can grow the show, improve the quality of the content and the production to get you better insights to stay ahead in your career.

Finally, do check out my bundle at [bundle.acg.com](https://bundle.acg.com) to get access to nine AI products for an entire year for free. This includes Dovetail, Mobin, Linear, Reforge, Build, Descript, and many other amazing tools that will help you as an AI product manager or builder succeed. I'll see you in the next episode.