TRANSCRIPT Episode 18 – Cameo Doran & Krystee Miller

Jonathan Bench: Today I'm joined by two accomplished technologists based in Utah. Cameo Doran is the owner of Cameo Labs, a product development partner for high-stakes software companies. With 20 years of building production systems for Fortune 500 clients, she helps Series A and B companies figure out what to build and then deliver world-class software that drives real business metrics.

Krystee Miller is the founder of Orb Agency, a product strategy and design agency focused on Web3 and AI, where she builds systems that make emerging technology practical, usable, and scalable.

Ladies, welcome to the podcast.

Krystee Miller: Thank you. Happy to be here.

Cameo Doran: Thank you for having us.

Jonathan: So let's start with Cameo. I want a little bit of your professional and personal arc—how you got into the technology space from where you started, and how you got to where you are now.

Cameo: I think my arc is kind of wackadoodle. I wanted to be a rock star. I spent a lot of my 20s trying to do that—got a small record contract, toured, did that whole thing. Then I got into my early 30s and had to settle down because the rock star thing didn't pan out.

I got into technology through a project management doorway because I'm naturally bossy. And then I just fell in love with technology. I fell in love with products—the art of product management—and how hard it is to figure out the right thing to build, and the interaction with humans.

That led to my 20 years in technology. I've mostly worked in agencies with big companies like Ford and Lockheed Martin, and smaller companies like LogMeIn and local companies here in Utah. I'm just such a fan of the possibility of innovation.

Jonathan: Okay, you can't drop that lore about being an aspiring rock star and having a record label without telling us what your instrument was. Was it voice or an instrument?

Cameo: My voice—and a tambourine, because at some point they always hand the singer a tambourine. But yeah, my voice. We never made it big, but we had a heck of a time doing it.

Jonathan: So fun. Topic for another conversation—that's going to be fun to talk about.

Cameo: Yeah.

Jonathan: And Krystee, what about you? How about your technology arc?

Krystee: Yeah. I got my start working for a children's animation studio. It's kind of funny—I was the production assistant, so I had to do a lot of odds and ends. One day they had me doing HTML and I was so confused. I didn't know what it was, so it started this whole chain of events.

I've worked in MLMs, helped build websites for distributors, and moved into subscription management software. Then I ended up in the agency world, working on so many different types of websites and marketing things.

Jonathan: Perfect. So the first topic for today is extremely important. It hit all of us at the same time: AI is everywhere. It's embedded in both of your worlds, and it's embedded in my world. It's accelerating development, automating complex analysis.

What have you seen in the AI space genuinely improving for users? And where do you think it's being applied in ways that create more confusion or fragility in the system? Either of you can jump in.

Krystee: Yeah. I think process speed. It's taking the mundane things you do daily and amplifying your output—taking the boring stuff and streamlining it. Pattern-heavy, error-prone work gets consolidated so you can focus on more rewarding things or more innovative tasks. That's one way I see it helping users.

Cameo: I think there are lots of different ways we're seeing AI come into our lives. Of course, everybody's using various chat applications. We're in the vibe-coding world—being able to build technology with a prompt. That's going to impact our lives and day-to-day in a big way.

I'll also say: I don't know that AI is always genuinely improving outcomes for users. There's data showing people are actually slower at some of their jobs, and coworkers are spending more time cleaning up each other's bad output.

I don't think we really know how to use the tools effectively yet. And I'm very pro-AI, so I don't want that to sound like I'm a Luddite. But I do think there's a lot of confusion and fragility happening, and organizations are struggling to understand how to make it make business better.

Jonathan: I'm glad you mentioned vibe coding, because that's an area I've played in. It lets non-technical professionals build applications using natural language. Those of us who can articulate things well think we can get to a better outcome.

Of course, I have clients doing the same thing and using AI to "play lawyer," and then I review everything. Instead of having no questions—because they didn't read my contract—they come back with 20 or 30 questions. Some are decent, and some are just missing context.

So on the vibe-coding side: companies have relied for years on IT departments or dev firms, but now they think they can do their own prototypes. Does it really help the workflow? And where are you seeing success—or semi-baked work product coming to you?

Cameo: I'm building vibe-code applications for all sorts of things—typically where there isn't an out-of-the-box tool I can go buy. I know business owners who have vibe-coded a replacement for HubSpot. I know that sounds like hype, but I've seen apps people built that are super focused on their exact needs.

A product like HubSpot is designed to fit the needs of a hundred thousand corporate customers. With vibe coding, you can get a tool that fits your needs—if you're good at describing what those needs are.

Krystee: That's the key: being able to say exactly what you want to build and knowing the constraints. A lot of people jump into these apps and start prompting away—"I want to build this cool thing"—and then they get stuck in a loop because they didn't think through X, Y, and Z. That's where teams are struggling.

The people who understand the planning stages are still the most important are the ones who actually get success at the end of the dev journey.

Jonathan: I always come back to this concept that planning and execution are where the magic is—startups, businesses, everything. You need good plans, people who can articulate them, and then carry them through. If it's not articulated well, people don't know what's happening. If nobody carries it to the end, you don't have the two core components that cut across industries.

Cameo: One of the most powerful things about vibe coding is the feedback is pretty instantaneous—and the friction is low to try something, realize it's not what you want, and rethink how you instructed it.

In traditional software development, you almost have to wait until somebody builds something—unless you're working with a firm that prototypes heavily, which is rare. The feedback loop is long. With vibe coding, you can speak a prompt, have a tool in less than an hour, and then rethink it. You can throw the first version away without feeling bad and try again. I do that all the time.

Jonathan: What's the fastest you've built something using vibe coding? How quick from ideation to "we've got a concept here"? Minutes? Less than 10?

Cameo: Yesterday at the gym, I was thinking about getting rid of the project management software we use internally. I started with Claude and talked through what I want—focus, what happens when I log in. I don't want to say we wrote a spec, but it helped refine my thoughts.

Then I plugged that into Lovable, and five minutes later I had something I was testing. It's going to have problems and won't be perfect, and it might take a couple days of sporadic iteration. But yeah—within minutes. And I was never at a computer. I was on the treadmill the whole time.

Jonathan: Krystee, what about you? Tell us about your workflow or creative process.

Krystee: I'm still sorting out the tools I like and prefer, and how in the weeds I want to be. I really enjoy seeing the code come out, making decisions, and feeling like I'm a big engineer.

Initially I'd jump into Lovable, run ideas through it, decide what I don't like. Then I realized the app I built wasn't going to scale the way I wanted it to with where Lovable was a few months ago—though it'd probably do great now.

So I started doing everything with Cursor and trying to understand what's happening. Now it's taking longer because I slowed down. It comes down to what you're trying to do: can the application support the long-term vision? Do you want to take payments? Sell it? Have people use it? Or is it an internal tool?

Some of the stuff I've built just for me is less than an hour. My eight-year-old built a game and was so excited sharing it with his friends. I'm like, "Okay—now we've got to add users." So it depends what you're building, which tools you like, and what you're comfortable with. The speed comes through that.

Jonathan: I'd love to hear each of you tell me a little bit about your favorite tools. I'm an outsider—I'm a lawyer. I learn about workflows as I help companies solve good problems or bad problems. I hadn't heard of Lovable until I saw one of you mention it on LinkedIn this week. I'm curious what's trending.

Krystee: I think Lovable is top of the line on the UI side. It gives you a visual component quickly. You can build fast. They have authentication, billing, and now AI tools you can integrate into your app. Instead of grabbing ChatGPT API keys, you can just use Lovable. I don't remember their term for it, but they're building an all-encompassing platform.

Cursor has been my favorite to play with because I like seeing the work happen and understanding what's in my codebase. If something breaks, I want to know what module broke and fix it rather than running AI scans and ending up in loops. That's my preference—other people might prefer the opposite. And I'll still use ChatGPT to explain an error to me. But yeah—those two are my favorites.

Cameo: I'm using a lot of different things, partially because we're a software development company, and I feel some obligation to stay on the cutting edge of what's working.

I really like Claude Code when I want to do something more in-depth that requires a level beyond Lovable. Lovable has a great user experience. It's easy for an entry-level person to jump in without needing to know much about how software is developed.

I've also really liked Replit and Base44. I've enjoyed playing with Base44—it's more complex and more detailed. Gemini and Google are releasing some great cutting-edge tools too. It's moving so fast—today is a world of difference from three or four months ago.

Jonathan: So great. We might revisit that, because I'm genuinely curious. I might text you once every few weeks: "What's trending now?" I've used Replit but hadn't heard of Lovable. I'm not even really in this space—I just like knowing what's going on, which is part of why I'm glad we get to talk today.

My audience is probably 50% business people and 50% lawyers. For lawyers, the role is changing rapidly with AI. We're expected to understand AI tools, help people evaluate vendors, sometimes build solutions—and we didn't go to law school for any of this.

What's the gap between what legal professionals think they need to know about this technology versus what would actually help them compete?

Cameo: That's a heck of a question.

Krystee: I know. I'm processing it.

Jonathan: Maybe we'll start small: when do lawyers suck in the equation? When do you hate having a lawyer in the room?

Krystee: I mean, I call you for everything, so I never mind. "Jonathan, can I do this?" But I think for lawyers, the best benefit will be streamlining repetitive work. Although that might take a hit on your bill since you bill hourly.

Jonathan: Billing is definitely part of the conversation—how we capture value while adding value. I have a great technology client who asked if I was using AI. I got defensive and said, "Yes, I'm using AI to capture efficiencies for you."

And he said, "I don't really care about that. I want to know you've read every word of anything you send me that I'm going to sign." So yes, use AI, but more importantly he wants to know I've signed off on it.

And back to the point—lots of clients never read my contracts. They're like, "No, you're the lawyer. That's what I pay you for."

Krystee: We'll read it if there's a problem later.

Jonathan: Yes.

Cameo: There's got to be an IP reckoning coming for all of AI. I think about it a lot—even in software development. If I can go into Base44 and say, "Make me a replica of Lucid," and it spits out a functional version in 10 minutes, I'm not trying to sell it, so there's probably not a blatant moment when I'm violating Lucid's IP.

But how are we decimating the bedrock of capitalism when I can replicate anything anyone's ever created, use it for my own benefit, and not pay them a cent?

Jonathan: We've come full circle to the 1990s—ripping off music with Napster and BitTorrent.

Cameo: Yes, we have.

Krystee: I saw a bunch of blank CDs at OfficeMax today and it was funny. I haven't seen those in a long time.

Jonathan: It does bring back memories. Okay—putting AI aside for a second: generally, as technologists, when do you find lawyers helpful?

Cameo: I'll speak on behalf of my clients. A lot of times I see resentment between development and legal—like legal is going to rain on everyone's parade at the eleventh hour.

I encourage people to talk to legal early. Know the guardrails and where they should be concerned. And coming back to AI, this is where internal systems could be cool—what if legal could create a "lawyer in the box" to flag moments where a feature set might have legal repercussions? That could prompt, "This is a great moment to call Jonathan."

Or maybe it triggers an alert to Jonathan that says, "They were in a meeting today talking about XYZ, which could put the company in a legal risk situation."

Jonathan: Krystee, what about you?

Krystee: I'm typically pretty risk-averse, so I want a lawyer in the room early. In past projects I've called you early to make sure I'm not crossing NDAs or conflicts.

I don't want to invest money and time into a product unless I have a full grasp on what's legally allowed. I think Cameo is right—people assume you're going to rain on the parade. I don't think that's the case. I think you're protecting the parade from falling apart when you're ready to launch.

To me it's critical, and it's an expense businesses need to prepare for.

Jonathan: In law school, they preach "learn to think like a lawyer." I felt like an outsider the whole time. It wasn't that I wasn't smart—I could keep up—but law school trains you to think in a certain way, especially around disputes.

It wasn't until months and years into my career that I started thinking: I do have a space here, it's just not talked about much in law school. We need lawyers on the side of the business saying, "Yes, let's push. I'll tell you where the line is. You can dance as close to it as you want."

I had a colleague who would lose sleep over clients' decisions. That was interesting to me. If you give good advice and they make a decision you wouldn't make, why lose sleep? It's not your business. It helped me understand we're all different, and we need different brain types and risk appetites. That diversity makes businesses functional.

Krystee: I'm that person who worries about other businesses' decisions. I love that you said that. That hit me. I worry when my clients make decisions I don't love. I worry about repercussions falling onto me. Maybe that comes from not understanding the line between my problem and their problem. That's interesting. I want to think about that more.

Cameo: Part of why I couldn't keep working in corporate is I was often in organizations making decisions I didn't approve of—out of my control—even if I was in leadership. It kept me up at night.

Jonathan: It's so interesting. Let's pause for a minute because we're getting close. Time goes by fast, but we can always do a follow-up. I'm going to push this episode to launch next week, right before Utah Tech Week, because I want to give you exposure for the platform you're launching or your event.

Tell us what the event is, who should attend, why it'll be great, and what they'll learn.

Cameo: The event is called Vibe Mania. It's Tuesday, February 3rd at 11:30 a.m. From 11:30 to noon we're eating lunch, and from 12:00 to 1:30 it's the vibe-coding workshop.

It's at Pellion. You can find the link on the Utah Tech Week website, and it'll have the address. It's been well received—we have a lot of people registered. If you're interested, sign up quickly. We think we'll have to cap it. We have about 60 already registered.

Our guarantee is you'll walk away after 90 minutes with a live application. We've joked we don't know what we'll do with the last 45 minutes because people will have something built.

Jonathan: I love that.

Cameo: It's sponsored by Lovable—they've been very generous. They'll give attendees a month of Lovable Pro so you can keep experimenting without hitting token limitations in the free version.

We'll share specific processes we use. We'll show how to refine an idea, and if people care about turning it into a business idea, what to do to validate it. We also want people to understand the limitations—there are definitely limitations. We don't want to overcommit, but that's the summary.

Jonathan: So everyone bring their laptop and build at the same time.

Cameo: Bring your laptop.

Krystee: I mean, you could probably do it from your phone too.

Jonathan: Or phone.

Krystee: You probably could.

Cameo: It's not as much fun on a phone. And honestly, no shade to Lovable, but the web app on your phone is kind of janky. Bring a laptop—we'll make it easy for people to work live. We'll walk around and help people troubleshoot and understand what to do.

Jonathan: Excellent. I'll pause now—any questions you want to talk about before we wrap up? Any burning thoughts you want to get in?

Cameo: I really liked the question about business leaders who relied on IT departments in the past—should they be prototyping their own workflows in vibe coding? I have some thoughts on that.

Jonathan: Great. Krystee?

Krystee: That's close to what I was thinking too—skills founders aren't utilizing right now.

Jonathan: Okay. So where's the line between prototyping yourself and bringing in experienced people. We'll do that. And I might ask your thoughts on the Utah tech scene

generally—where it's been and where it's going. Then the last one: what you'd be doing if you weren't doing what you're doing now.

So let's turn to vibe coding within an existing organization—startup or established company. A lot of business leaders are used to relying on IT departments to make decisions and decide what gets used. Right now we're all using technology IT has no idea we're using. In that context, how much should business leaders and entrepreneurs be prototyping their own stuff? Where do you encourage them to play and learn before bringing in experts?

Cameo: I'm conflicted. I think any leader—and any lawyer—should understand these tools and see what's possible. But if you're a lawyer, and your value is in billable work or helping clients, I don't know that you should spend most of your time building applications, even if it's easy—three or four hours to refine a workflow. I'm conflicted that it's the best use of time.

So I encourage people: get your hands wet and get familiar, but also ask whether you're bringing value if you go down developer rabbit holes.

Krystee: I think that rabbit hole catches everyone—across departments. There are times I'm going through slide deck edits and everyone sends me different interpretations from their AI model, and I'm like, "Everyone, just read it. Let's just read it. It's fine. We can still make decisions."

But personally, I don't want to work with another founder who hasn't tried to build something before. Seeing the development process and understanding decision-making is a skill set you should have. It saves money, headaches, and fights over basic questions founders haven't thought of because they haven't been in that headspace.

So I'd encourage anyone: if you want to build an app, don't hire an expensive team until you've tried to build something yourself. It can be small—anything—just so you understand the process.

Cameo: I like that.

Jonathan: Super interesting. Thank you both.

What's your take on the Utah tech scene? How's it gone, how's it going, where is it going? What are we doing well, what can be improved, and is it a great place to live and build?

Krystee: I think it's amazing. The community is phenomenal. You can reach out to your LinkedIn network and find almost anybody to help with a problem. I have a lot of people I trust and could connect with if I needed support.

As far as what's being built, I see founders leaning into AI tooling to understand their startups and scale faster. Time will tell if it works. But overall, the community is strong and it's a great place to build.

Cameo: I feel the same way. I've loved working in this community. People are open and willing to help. As a woman, I'd love to see more diversity. I've been in tech 20-plus years, and it's pretty much the same as it's always been.

I don't know if that's a choice women are making or if they don't feel there's space. I don't think there's anyone to blame necessarily. But I am disappointed when I see founding team after founding team and there's never a woman included. The numbers are clear: diverse teams tend to get better outcomes—not always, but it improves your odds.

Jonathan: Thank you so much. Last question—I prepped you briefly but didn't give you much time to think. If you weren't doing exactly the role you're doing right now, what would you be doing? This could be "I don't have to work anymore," or "if I retooled professionally, here's what I'd do." Cameo, the band takes off again—that kind of thing.

Krystee: I wanted to be a child psychologist. I'm a child of trauma—I went through a lot as a kid—and I've always wanted to find other ways to help kids. I think that's also why I was drawn to product design—there's a lot of psychology mixed in. But yeah, I'd go into psychology.

And traveling—I want to take my kids to Tokyo someday.

Jonathan: Tokyo. Nice.

Krystee: His dream, not mine. I think it'd be great, but that's his big goal.

Cameo: It's a good goal. Good for him.

Krystee: Yeah, he's learning Japanese, so we'll see.

Cameo: Well, to quote Dr. Dre: music is a young man's game. But if I had money and didn't have to work, I'd be investing in other artists.

It's a dangerous time in music because AI could decimate the ability for new artists to get heard—especially if platforms like Spotify overemphasize their own AI-created music, which they probably would, versus helping new artists get in front of people. And I love new art.

Also, I would take my kids to Tokyo.

Jonathan: So fun. Well, thank you both. This has been wonderful. I love that we have an in-person relationship. It's fun to connect online and produce content together that I hope

helps people think more. I learn every time I have these conversations, and you two have helped me a lot. I really appreciate it.

Cameo: Thank you for having us. Appreciate the invite.