AI360

Exploring the impact of Gen AI in software development

Rohan Gupta: Hello, everyone, and welcome to another episode of AI360. I'm your host, Rohan, and I'm really excited by our guest today. It's my longtime friend, Julie. Fun fact: We actually started out as senior consultants over a decade ago, but Julie has actually been at the firm a little bit longer.

Julie Shen: You're dating me, Rohan!

Rohan Gupta: Do you want to do a quick intro?

Julie Shen: Sure. My name is Julie Shen. I'm a principal in Deloitte Consulting's Technology, Media and Telecom practice, and I help clients explore the possibilities of AI as it relates to software development. Happy to be here.

Rohan Gupta: Yeah. Thanks so much again for coming, Julie. I'm really excited about this topic because we've talked a lot about the impact of generative AI on businesses, but we haven't really touched on the developer impact, so particularly excited about this one. So maybe if you could just start by telling us what are some of the types of tools that we're seeing in this space, and what impact have they had?

Julie Shen: Yeah, absolutely. So there are two user groups that we need to consider when it comes to thinking about the impact of gen AI on software development. The first is professional developers. One recent study of US-based developers found that 92% were already using AI coding tools in or outside of work. Seventy percent of developers say that these tools give them an advantage, citing benefits around better code quality, faster completion time, and quicker incident resolution. So clearly, professional developers are already embracing gen AI. Coding companions and other tools reduce the time spent on repetitive tasks, such as test-case creation, updating documentation, or fixing bugs, to allow developers to focus on more complex and creative programming challenges. The second group is actually users who do not have a background in software development. So, what if you could prototype an app without ever having to learn how to code? Gen AI, when combined with low code/no code, has a democratizing effect. Nontechnical users can write user stories, build apps, and test performance with just natural language prompts.

Rohan Gupta: That's really cool. I guess I could tell people I'm a coder now just because of gen AI. So given this proliferation of gen AI and the tooling all across the ecosystem, what are some of the key considerations to evaluate whether a particular tool or technology is right for an organization?

Julie Shen: Yeah. So first, start with your user—the developer. What are the coding languages they most frequently use? What integrations are most critical, and what routine tasks are taking up most of their time? Second, understand the limitations. This is still an evolving space. AI coding tools have limitations around accuracy, control, security, and more. Do your research to understand the risks. And lastly, think through the supporting investments. To truly realize the full upside, development teams will need new skills, new processes, and new adjacent technologies to integrate gen AI tooling into their workflow. For example, most companies will need to upskill development teams in areas like machine learning, data science, and prompt engineering to make the best use of these tools.

Rohan Gupta: Yeah, I love that it's almost like you could use generative AI to get better at using generative AI. We actually heard that a few times before. So I guess what's really interesting about the particular space that you're talking about, Julie, is that software developers are essentially the people that might be automating themselves out of a job. How do developers feel about generative AI?

Julie Shen: Yes, that's a really interesting question, Rohan. We've actually worked with some clients to explore this exact issue, and I'll share some learnings from one project in particular. So, most developers agree that AI will probably never completely replace humans in software development. However, there's an interesting difference in sentiment between developers with more experience and those with less. The seasoned developers are more skeptical. Only about a third trust the accuracy of AI-generated output. This group also feels less threatened by AI. Fewer of them believe that AI will replace them in the future. The less experienced developers are more trusting. Almost half believe in the accuracy of AI-generated output. This is also the cohort that feels more threatened by AI, more likely to be replaced. And if you think about it, this finding is actually intuitive. The more skeptical you are of a technology's efficacy, the less likely you think it will replace you, the human. But regardless of experience level, developers agree that those with better understanding of AI will be the ones better equipped for the future.

Rohan Gupta: That makes a ton of sense. I guess, as we look ahead and you're advising developers, what are some of the no-regret moves that they can make to ensure the success—and to ensure their success, their company's success—as AI explodes around them?

Julie Shen: Yeah, here are my top three. So number one, focus on the human, not the AI. Remember, these are only tools. They're meant to support your developers. Make sure your developers are the hero of the story. Number two, don't forget about the soft skills. Skills such as communication and collaboration, industry and domain expertise, and creative problem-solving that all reside in a great developer today will only be more important in the future as these tools blur the boundaries of technical and nontechnical teams. Continue investing and training your developers in these skills—in these soft skills, not just the AI ones—to unlock their full potential. And number three, lead with trust. Communicate with transparency to your developer teams, adjacent teams, even your customers and partners, around which use cases you'll focus on and where you are in the maturity curve, as well as how you'll mitigate potential risks. Invite these groups, especially your developers, to chart your journey together. regulatory bodies, quite frankly, require and expect from the full ecosystem of content creation.

Rohan Gupta: That makes sense. I love the idea of making the developer the hero of the story. I'm going to steal that, if you don't mind.

Jule Shen: Of course, any day.

Rohan Gupta: Julie, thanks so much for joining. This has been one of the best six minutes I have been a part of. We'll definitely have you back in the future. And for all of our listeners, you know, we're going to be back with some more. So please share this with your friends, and give some other episodes a listen as well. Thanks so much.

Jule Shen: Thank you.