Interview Transcript – Participant: [Anonymized]

Date of Interview: 20 January 2025

Mode of Interview: Online (Zoom)

Interviewer: Anonymized

Participant Role: Software Engineer

Duration: 22:14 Min

Consent Obtained: Yes

All personally identifiable information has been anonymized.

---

**Interviewer:** Hello

**Participant:** Hello!

**Interviewer:** How are you?

**Participant:** I’m good, thank you.

**Interviewer:** Yeah, I'm okay too.

**Interviewer:** How many years of software development experience do you have?

**Participant:** I have more than four years of software development experience.

**Interviewer:** Do you have any idea what a code clone is?

**Participant:** Yes, I do.

**Interviewer:** Are you familiar with clone types?

**Participant:** Yes, I know about syntactic and semantic clones.

**Interviewer:** Are you aware of AI-based tools like ChatGPT, Gemini, Claude, etc.?

**Participant:** Yes, I’m very familiar with those AI tools.

**Interviewer:** That's great. Do you use them to generate code?

**Participant:** Yes, I do use them to generate code.

**Interviewer:** Do you believe software engineers are using these AI tools in their daily development work?

**Participant:** Yes, I think most software engineers are using them to help with development.

**Participant:** I started using Copilot around six months ago.

**Interviewer:** Okay. So, how did you come to know about it?

**Participant:** A colleague of mine recommended it, and I decided to give it a try.

**Interviewer:** Great. So what's your primary use case?

**Participant:** Mostly for generating boilerplate code and sometimes fixing bugs.

**Interviewer:** Do you use it daily or occasionally?

**Participant:** I’d say occasionally, maybe a few times a week.

**Interviewer:** Can you describe one situation where Copilot was really helpful?

**Participant:** Sure. Once I was working on a REST API, and Copilot suggested the entire controller logic. it saved me at least an hour.

**Interviewer:** Interesting. Did you ever face any issues with the suggestions?

**Participant:** Yeah, sometimes it gives outdated or insecure code, so I double-check.

**Interviewer:** That’s a good point. Do you think Copilot influences your learning or coding style?

**Participant**: A little bit, yeah. I’ve learned new libraries from it, but I try not to rely on it too much.

**Interviewer:** Thanks a lot, that was really helpful.

**Interviewer:** I’ll ask you some more questions based on that. We're currently working on a study titled \*“Are classical clone detectors good enough for the AI era?”\* As part of the study, we are interviewing developers like you who use large language models (LLMs). Before we dive into your personal insights, we’ll look at a few examples together and then discuss them further.

---

**Demonstration Phase – Summary**

During this phase, the interviewer presented participants with several code snippets: one set written by humans (sourced from Stack Overflow) and another set generated by AI tools, both designed to perform the same functionality. Participants were asked to review the snippets and identify which ones were AI-generated and which were human-written.

**Key Insights:**

- The participant couldn’t determine all the AI-generated code. (after removal of comments)

- The participant observed that AI-generated code appeared more structured compared to human-written code.

- The participant agreed, noting that different humans write code in different ways and naturally vary in their coding structure.

- The participant also referenced platforms like Codeforces and LeetCode, where developers often write distinctly different solutions for the same problem.

- The participant also thinks apart from common solutions; human tends to write more syntactically different codes.

- The participant also thinks that classical clone detectors might be able to detect clones regardless of whether it is human-generated or AI-generated, but finding which clone is generated or human-written. But the participant was not sure about semantic clones; it might be hard for the clone detectors to detect the AI-generated semantic clones.

- Later in the discussion, the participant mentioned that the complexity of the problem could influence the outcome—if the problem is complex, the code generated by AI may also be more complex. However, they noted that having a human review or monitor the output could help manage this complexity effectively.

-The participant also highlighted that AI-generated code generally adheres well to coding standards and is often of high quality.