Hi everyone,

Types of Code/Peer Reviews: Applications and Use Cases

Code reviews play a crucial role in improving code quality, fostering collaboration, and ensuring consistency in software projects. Two common types are pair programming and over-the-shoulder reviews, each suited for different scenarios.

1. Pair Programming: In pair programming, two developers work together at the same computer. One writes the code (the “driver”) while the other reviews it in real-time (the “observer”). This approach is highly collaborative, allowing immediate feedback and shared learning. It’s especially useful for solving complex problems or training new team members. For example, when tackling a challenging feature or debugging an issue, pair programming can speed up the process while ensuring the solution aligns with best practices. Here problems that may occur are; increased time and cost, skill imbalance, potential for fatigue, team compatibility issues, and it’s not suitable for all tasks, therefore limited

2. Over-the-Shoulder Reviews: In this informal review style, a developer explains their code to a reviewer who provides feedback on the spot. It’s quick and straightforward, making it ideal for smaller teams or less critical changes. For instance, if a team member implements a minor fix, an over-the-shoulder review can verify the update without the need for a formal process. This however is also the problem, it lies in their informality and lack of documentation and can lead to problems such as, Missed Opportunities for better more comprehensive input, Limited Scalability, Inconsistency, and others.

Both approaches promote collaboration, they differ in intensity and use cases. Pair programming works best for long-term benefits like skill-building and solving complex tasks, whereas over-the-shoulder reviews are perfect for quick iterations or minor changes.

<https://axolo.co/blog/p/types-of-code-reviews-maximizing-your-code-quality>

<https://graphite.dev/blog/types-of-code-review>