



## 10 Hacks to Mitigate Technical Debt in Low Code Application Development

---

Technical debt is the cost of taking shortcut solutions that are quick and easy to implement but may not be the best long-term solution, leading to potential maintenance difficulties in the future.

Technical debt can pose a major challenge for software developers, especially when working with low-code platforms. While it may not always be easy to quantify its exact impact, managing technical debt effectively can help reduce costs and improve overall project success.

Here are ten best practices for identifying and addressing technical debt in low-code environments:



**Solution design:** Define robust security standards, a well-designed domain model, seamless integration, and a performance optimization strategy.



**Involve developers in the design phase:** Empower organizations to leverage the benefits of rapid development while maintaining a high level of quality, security, and customization.



**Manage version control (branching strategy):** Low-code platforms that support version control can help you track changes to your code over time, and they can also help you to roll back changes if necessary.



**Prefer reusable/pre-built/tested components:** Save time and effort with reusable components that can improve the consistency of your code.



**Regularly conduct code reviews and code quality:** Schedule regular code reviews to identify potential technical debt issues and address them promptly.



**Adopt best practices:** Stay informed about the latest developments in low-code technology and best practices to ensure your applications remain relevant and efficient.



**Perform unit testing:** Perform unit tests to verify the functionality of the low code application, as you make changes to it.



**Use performance monitoring tools:** Implement performance monitoring tools to identify and address bottlenecks or performance issues early on.



**Prioritize technical debt resolution:** Allocate time and resources to tackle technical debt proactively instead of letting it accumulate over time.



**Continuous refactoring:** Improve the design and architecture to simplify application maintenance and extensibility, and optimize performance and scalability, which can reduce the risk of technical debt.



## About Indium

Indium is an AI-driven digital engineering company that helps enterprises build, scale, and innovate with cutting-edge technology. We specialize in custom solutions, ensuring every engagement is tailored to business needs with a relentless customer-first approach. Our expertise spans Generative AI, Product Engineering, Intelligent Automation, Data & AI, Quality Engineering, and Gaming, delivering high-impact solutions that drive real business impact.

With 5,000+ associates globally, we partner with Fortune 500, Global 2000, and leading technology firms across Financial Services, Healthcare, Manufacturing, Retail, and Technology—driving impact in North America, India, the UK, Singapore, Australia, and Japan to keep businesses ahead in an AI-first world.

### USA

Cupertino | Princeton  
Toll-free: +1-888-207-5969

### INDIA

Chennai | Bengaluru | Mumbai  
Hyderabad | Pune  
Toll-free: 1800-123-1191

### UK

London  
Ph: +44 1420 300014

### SINGAPORE

Singapore  
Ph: +65 6812 7888

[www.indium.tech](http://www.indium.tech)



For Sales Inquiries  
[sales@indium.tech](mailto:sales@indium.tech)



For General Inquiries  
[info@indium.tech](mailto:info@indium.tech)

