TECHNOLOGY ASSESSMENT GUIDE

Sponsored by: Google

Developers have widely adopted AI coding assistants, providing them with real-world knowledge and leading to an increasingly sophisticated awareness of the most important features and capabilities.

Generative AI in the Software Development Life Cycle: An IT Leader's Guide

September 2024

Written by: Michele Rosen, Ph.D., Research Manager, Open GenAI, LLMs, and the Evolving Open Source Ecosystem

Introduction

AI coding assistants have gone mainstream, and organizations worldwide are adopting these tools to supercharge the productivity and innovation of the development teams building and maintaining their digital infrastructure. From the summer of 2023 to the summer of 2024, 90% of developers used a coding assistant to help them develop production-grade digital solutions, according to IDC's September 2024 *U.S. Generative AI Developer Survey*.

Now that developers have moved beyond experimentation and integrated these tools into their workflows, they have a much better sense of what they need from an AI coding assistant, and they are evaluating the available tools from the perspective of experience. According to IDC's research, developers want coding assistants to integrate with their existing toolsets and to produce high-quality results with equally high levels of security, privacy, and data protection. Given the current state of the art, achieving these results requires an AI coding assistant with options for customization based on the organization's code, documentation, and best practices.

With recent improvements in the underlying foundation models, IDC predicts that by 2027, AI will be capable of automatically generating code to meet functional business requirements for 80% of new digital solutions in development and early deployment. During the next three years, developers will increasingly interact with a team of AI assistants, making it easier for less experienced developers to get up and running and for more experienced developers to focus on more challenging tasks, including incorporating intelligent features into their digital solutions.

AT A GLANCE

KEY STATS

Among global organizations, 17% have already introduced several applications/services enhanced with generative AI into production, with another 38% investing significantly in GenAI (source: IDC's *Future Enterprise Resiliency and Spending Survey,* April 2024). According to the survey, GenAI is currently having the greatest impact on technology decision-making, beating financial/operational systems, customer engagement/experience/support, and other business areas.

Generative AI in the Software Development Life Cycle

One of the complexities of the generative AI (GenAI) landscape is that there are so many potential use cases for the technology. IDC has created GenAI use case taxonomies for 13 business function areas, including software development (see *Generative AI Use Case Taxonomy: The Software Development Function*, IDC #US51949424, March 2024). The taxonomy includes 25 GenAI use cases for the software development life cycle in five subsegments:

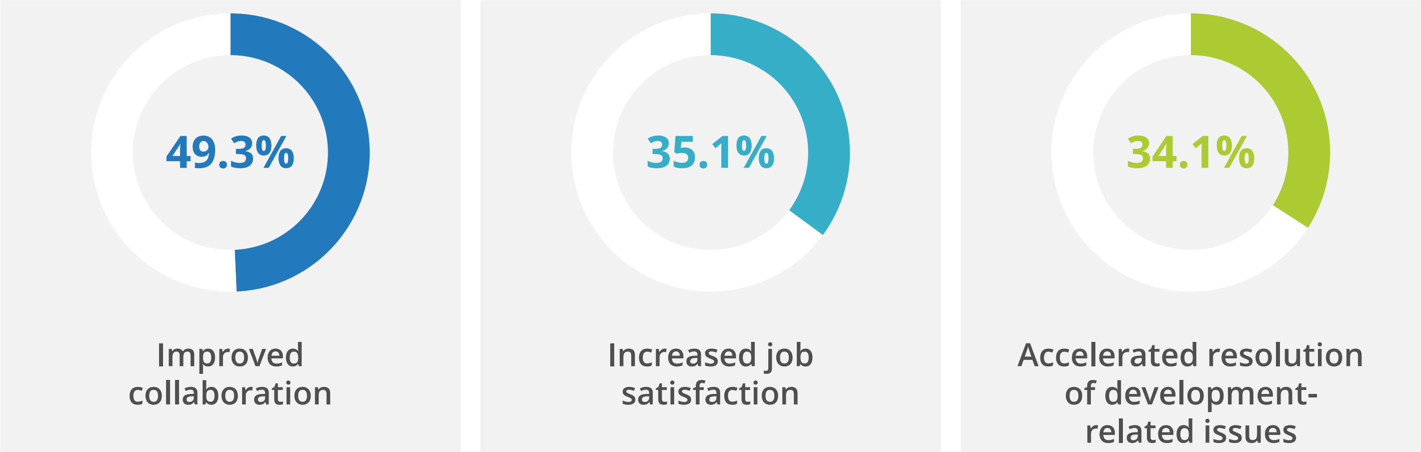
- **Planning:** Scope management, requirements documentation, resource allocation, task management, and collaboration
- **Development:** Application development, application modernization, application maintenance and operations, documentation and explanation, and UI/UX design and multimedia asset generation
- **DevOps:** Data preparation, pipeline efficiency optimization, model training and tuning, infrastructure as code, and value stream management
- **Quality:** Code quality analysis, testing, risk assessment and management, performance optimization, and model performance benchmarking
- **Safety:** Threat modeling, identification and remediation, monitoring, security policy and generation, compliance, ethics management, and governance

To implement these use cases and others, such as test coverage analysis and risk analysis, organizations are adopting a plethora of new tools, among which AI coding assistants are the most mature, in part because of their widespread adoption and their general utility for many of these use cases. Developers are finding that coding assistants can improve development team productivity in various ways, from explaining existing code to generating tests to improving code quality (refer to Table 1).

More than three-quarters of developers responding to IDC's 2024 *GenAI Developer Trends Survey* reported productivity increases as a result of the use of AI coding assistants. Interestingly, though, when asked about the top benefits of these tools, they cited other factors more frequently than productivity (see Figure 1).

FIGURE 1: ***Top Benefits of AI Coding Assistant to Developers***

Q ***Thinking about AI coding assistants you have used in the past 12 months, what are the top 3 benefits of coding assistants to developers and the developer experience?***



n = 211

Notes:

Multiple responses were allowed.

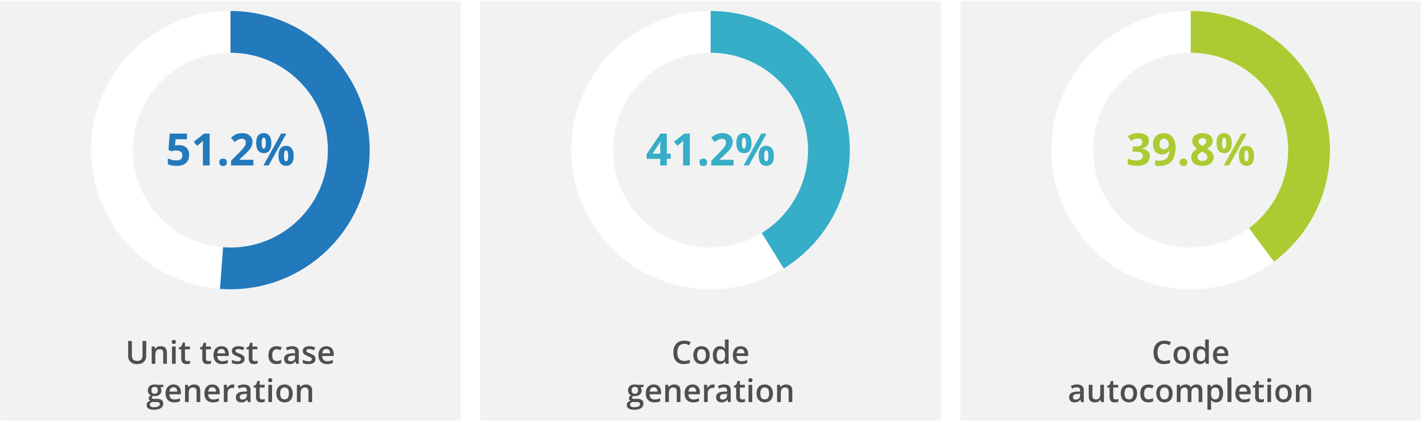
Use caution when interpreting small sample sizes.

Source: IDC's U.S. Generative AI Developer Survey, September 2024

Figure 2 shows that developers are using AI coding assistants for unit test case generation even more frequently than for code generation.

FIGURE 2: ***How Developers Are Using AI Coding Assistants***

Q Thinking about AI coding assistants you have used in the past 12 months, what are the top 3 coding assistant capabilities you have used?



n = 211

Notes:

Multiple responses were allowed.

Use caution when interpreting small sample sizes.

Source: IDC's U.S. Generative AI Developer Survey, September 2024

This IDC Technology Assessment Guide explores the leading use cases for AI-assisted development and DevOps and the critical questions that IT leaders must consider when adopting this technology.

Key Benefits and Use Cases for AI Coding Assistants

AI coding assistants can improve productivity for all developers by eliminating repetitive tasks, answering questions, and providing an informed interlocutor for brainstorming and learning. These capabilities help developers by reducing cognitive load and the need for context switching. Heavy cognitive load leads to errors and overgeneralizations, particularly for more difficult tasks. Research shows that multitasking, such as switching to a browser to look up information while coding, reduces productivity. By generating and explaining code, AI coding assistants, particularly those incorporated directly into the developer's existing IDE, can help developers reduce or eliminate the need to leave their IDE to look up code or documentation.

AI coding assistants can also help developers by automating tasks in real time. Organizations are already using process automation tools to automate tasks throughout the software development process, but GenAI coding assistants empower developers to offload tasks on an ad hoc basis during the development process where appropriate, decreasing human errors and accelerating processes.

AI coding assistants also embody the human-in-the-loop (HITL) approach to AI, which emphasizes solutions that assist and augment human capabilities instead of replacing them. HITL-oriented tools are more likely to produce aligned output because they run through an iterative development process in a feedback loop.

AI-Assisted Development

GenAI coding assistants can recommend full functions and blocks of code in response to natural language or by evaluating context as the developer types. These tools can also explain code in natural language, making it much easier to update existing code and providing a learning tool for less experienced developers. Because of their versatility, AI code assistants can improve developers' productivity throughout the software development life cycle:

- More than half of professional developers (52.1%) expect GenAI to augment UI design/development (source: IDC's *U.S. Generative AI Developer Survey,* September 2024).
- Interacting with a code assistant through natural language will empower line-of-business developers to create digital solutions that they could not have otherwise created, particularly in combination with no-code development environments.
- Infrastructure developers can also benefit from GenAI coding assistants. For example, almost half of developers (49.3%) expect GenAI to augment the process of deployment automation (source: IDC's *U.S. Generative AI Developer Survey,* September 2024).
- Data engineers, scientists, and analysts can benefit from GenAI coding assistants by adding this technology to their existing tools and development interfaces, including notebooks.

By augmenting existing skill sets, GenAI coding assistants can facilitate collaboration, upskilling, and cross-skilling, all of which contribute to productivity. Developers are particularly aware of the value of these tools for improving collaboration (refer back to Figure 1).

To evaluate GenAI coding assistants, decision-makers should determine the most important considerations for their key enterprise use cases and then evaluate vendor offerings based on those questions (refer to Table 2).

AI-Assisted DevOps

Software Testing

Enterprises have been using AI to define and execute test cases for several years. However, despite advances in test automation, DevOps teams continue to struggle with software testing, which impacts application quality and code release deployment frequency. As a result, more than half of DevOps teams are expanding, using, or piloting GenAI in their DevOps pipeline, with an additional 21.9% planning to adopt it within a year (see *DevOps Practices, Perceptions, and Tooling Survey, 2024: Automation, Observability, GenAI, and Purchasing Trends,* IDC #US52321924, June 2024). According to the survey, the top 5 DevOps use cases for GenAI are security testing, anomaly detection and monitoring, configuration management, code review, and test generation.

DevSecOps

Enterprises continue to face a growing application threat landscape. As a result, many companies have adopted DevSecOps practices that involve shifting security further to the left — as far as the planning and design phases — so that all enterprise applications are secure by design. Key priorities include ensuring security across multiple cloud environments, security policy management, and software supply chain security (source: IDC's *DevSecOps Survey,* January 2023).

However, IT leaders say that developers need more security knowledge and training to facilitate DevSecOps practices. To address this skills gap, AI coding assistants can help developers and DevOps professionals model threats, analyze scan results, detect anomalies, recommend remediation, and generate management reports.

AI coding assistants can also help developers and DevOps professionals find the information they need to make decisions. Foundation models are exceptionally good at finding patterns in and summarizing unstructured data, conversations, and policies. Conversational assistants that use models tuned on custom enterprise data can query logs and scans to guide remediation, explain security scan results and attack paths, and make security recommendations.

Next Steps

IT leaders seeking to leverage GenAI in the software development life cycle can take the following steps to move forward:

- Identify use cases and evaluate their applicability to pain points in the enterprise development process. IDC has created a GenAI use case taxonomy encompassing productivity, business function, and industry use cases. IT leaders can use IDC's GenAI use case taxonomy to begin exploring the use cases of greatest value to their organizations (refer to Table 1).
- Evaluate the risks of using GenAI for the identified use cases. Determine how the following would impact the use case:
  - Hallucinations
  - Unvetted training data
  - Legal challenges to training data use
  - Risk of data leakage
  - Data bias
- Identify key tooling, staffing, and resourcing required to operationalize and manage GenAI tools for each use case.
- Enable iterative experimentation and adoption by providing a sandbox environment with the appropriate GenAI capabilities for each prioritized use case.
- Determine which use cases call for off-the-shelf GenAI solutions and which require the use of enterprise data and additional governance and security capabilities.
- Evaluate the enterprise's data resources to determine the most useful data for few-shot training and fine-tuning.

Worksheet Section

Organizations can evaluate potential GenAI development and DevOps use cases and GenAI developer tools/vendors based on the frameworks provided in Tables 1 and 2.

TABLE 1: ***GenAI Development and DevOps Use Case Assessment***

GenAI Use Cases for Software Development and DevOps	How Well Does Your Organization's Current Solution Address This Use Case? (Very Well, Well, Somewhat Poorly, Poorly)
Development: Application development	
Development: Application modernization	
Development: Application maintenance and operations	
Development: Documentation and explanation	
Development: UI/UX design and multimedia asset generation	
DevOps: Data preparation	
DevOps: Pipeline efficiency optimization	
DevOps: Model training and tuning	
DevOps: Infrastructure as code	
DevOps: Value stream management	

Note: For more information, see Generative AI Use Case Taxonomy: The Software Development Function (IDC #US51949424, March 2024).

Source: IDC, 2024

TABLE 2: ***Evaluation Criteria for GenAI Coding Assistants***

	Response
Does the tool integrate with developer IDEs?	
Can the tool be customized or tuned using enterprise code and data?	
How accurate is the generated code for a range of use cases?	

	Response
Does the vendor provide indemnification for the code produced by the tool?	
How fast is the tool? (Latency above 100ms can interrupt developer flow.)	
Does the tool provide appropriate developer experiences for all personas related to your key use cases?	
What is the maximum context window for the underlying model?	
On what kind of data was the underlying model trained?	
Does the tool provide enterprise-level monitoring and control over developers' use of GenAI features?	
Are both the developer's input and the model's output supplemented by complementary tools, such as security scanning or software quality tools?	
Is the tool capable of scaling to meet all potential use cases?	
If the training data included public code, is the source license of the code identifiable?	
If line-of-business users or junior developers will use the tool, is it integrated with no-code or low-code development tools to further increase productivity?	

	Response
How does the vendor ensure the security and privacy of enterprise data?	
Does the vendor have tools to manage the enterprise data required to fine-tune models for relevant use cases?	
Does the vendor provide tools to ensure responsible AI governance?	
How has the vendor addressed generative AI–specific risks, such as hallucinations and bias?	
Does the vendor provide observability and governance features such as prompt or output logging?	
How does the vendor ensure the quality of generated code?	
Does the vendor enable integration of generative AI tools with other tools in the software development life cycle?	

Source: IDC, 2024

About the Analyst

Michele Rosen, Ph.D., *Research Manager, Open GenAI, LLMs, and the Evolving Open Source Ecosystem*

Michele Rosen is research manager for IDC's Open GenAI, LLMs, and the Evolving Open Source Ecosystem practice. Dr. Rosen's research focuses on the open source ecosystem and the emerging role of open source communities in delivering and governing GenAI software and large language models. Open source tools such as Jupyter, PyTorch, and TensorFlow have been integral to the development of generative AI, and open LLMs are beginning to play a significant role in fostering innovation and democratizing access to powerful natural language processing tools.

MESSAGE FROM THE SPONSOR

Google Gemini Code Assist offers AI-powered software development assistance to help developers build applications with higher velocity, quality, and security. It is available in many popular IDEs, such as Visual Studio Code, JetBrains IDEs, Cloud Workstations, and Cloud Shell Editor, and it supports 20+ programming languages.

Powered by Google's Gemini models with the industry's largest context window, Gemini Code Assist provides high-quality AI assistance with the ability to address very complex coding tasks across your entire repository. It comes with robust enterprise-level security, privacy, and data protection, and it offers code customization so that enterprises can train Gemini Code Assist by connecting to their private code base, wherever it lives.

Learn more at cloud.google.com/products/gemini/code-assist, or sign up for the Gemini Code Assist enterprise pilot program.

IDC Custom Solutions

The content in this paper was adapted from existing IDC research published on www.idc.com.

IDC Research, Inc.
140 Kendrick Street
Building B
Needham, MA 02494, USA
T 508.872.8200
F 508.935.4015
Twitter @IDC
idc-insights-community.com
www.idc.com

This publication was produced by IDC Custom Solutions. The **o**pinion, analysis, and research results presented herein are drawn from more detailed research and analysis independently conducted and published by IDC, unless specific vendor sponsorship is noted. IDC Custom Solutions makes IDC content available in a wide range of formats for distribution by various companies. A license to distribute IDC content does not imply endorsement of or opinion about the licensee.

External Publication of IDC Information and Data — Any IDC information that is to be used in advertising, press releases, or promotional materials requires prior written approval from the appropriate IDC Vice President or Country Manager. A draft of the proposed document should accompany any such request. IDC reserves the right to deny approval of external usage for any reason.