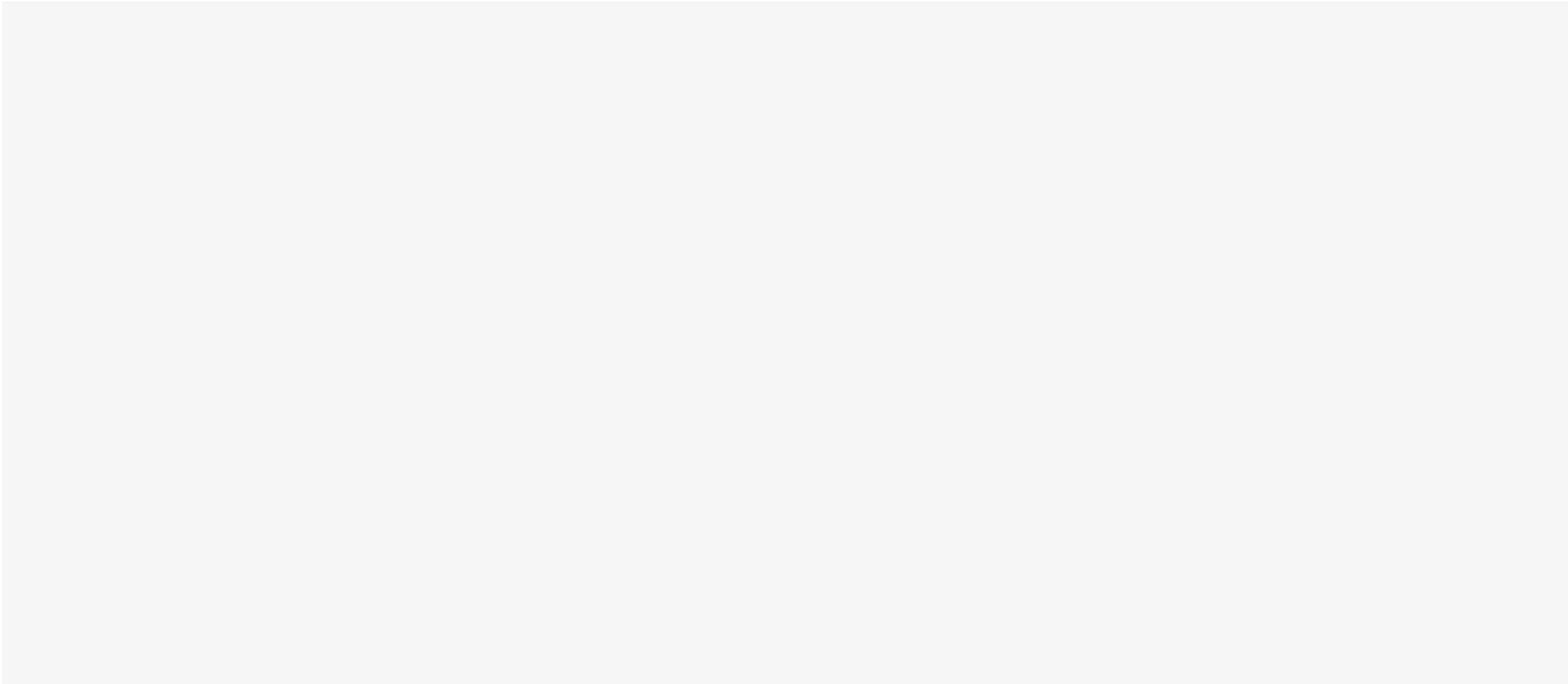


Turbocharging software with Gen AI

How organizations can realize the full potential
of generative AI for software engineering

Table of Content



Executive Summary

Organizations are reaping multiple benefits from leveraging generative AI for software engineering.

- The leading benefits for organizations are enabling more innovative work, such as developing new software features/services (observed by 61% of surveyed organizations), improving software quality (49%), and increasing productivity (40%).
- Organizations using generative AI have seen a 7–18% productivity improvement¹ in the software engineering function as per early estimates. This is highest for specialized tasks such as coding assistance² (34% as the maximum potential for time savings with 9% on average) and creating documentation (35% as the maximum potential for time savings with 10% on average). This research analyzed time savings in various software engineering tasks using generative AI tools and not cost savings which can be significantly different.
- Organizations are utilizing these productivity gains on innovative work such as developing new software features (50%) and upskilling (47%). Very few aim to reduce headcount (4%).
- Generative AI is having a positive impact on software professionals' job satisfaction.
 - 69% of senior software professionals and 55% of junior software professionals report high levels of satisfaction from using generative AI for software.
 - 78% of software professionals are optimistic about generative AI's potential to enhance collaboration between business and technology teams.

Executive Summary

Generative AI adoption is at an early stage but will accelerate sharply.

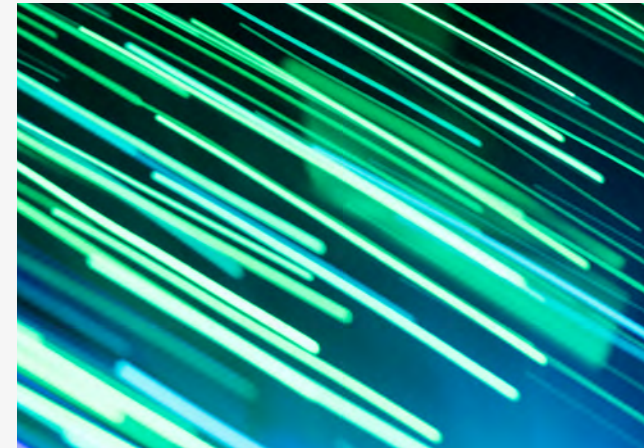
- Adoption of generative AI for software engineering is still in its early stages, with 9 in 10 organizations yet to scale.
 - 27% of organizations are running generative AI pilots, and 11% have started leveraging generative AI in their software functions.
 - Three in four (75%) large organizations (annual revenue greater than \$20 billion) have adopted (piloted/ scaled) generative AI compared to 23% of their smaller counterparts (annual revenue between \$1–5 billion).
- Adoption (including pilots) is expected to increase significantly in the next two years from 46% of software workforce using generative AI tools today (for any kind of training, experimenting, piloting, and implementing, with authorized or unauthorized access) to an estimate of 85% in 2026.
- Generative AI is expected to play a key role in augmenting the software workforce with better experience, tools and platforms, and governance (assisting in more than 25% of software design, development, and testing work by 2026).
- Coding assistance is the leading use case, but generative AI also finds applications in other software development lifecycle (SDLC) activities (test case generation, documentation, code modernization, UX design assistance, etc.)
- Most use cases have yet to be adopted by a majority of organizations (39% are focusing on coding assistance and 37% on UX design assistance as top adopted use cases).

Executive Summary

Lack of foundational prerequisites and unofficial usage of generative AI pose significant functional, security, and legal risks.

- 27% of organizations have the platforms & tools, and 32% have talent prerequisites in place, to implement generative AI for software engineering.
- Over 60% lack governance and upskilling programs for generative AI for software engineering.
- Of those software professionals who use generative AI, 63% use unauthorized tools.
- Nearly a third of the workforce is self-training on generative AI for software as less than 40% of employees are receiving training from their organizations.

- Using unauthorized tools without proper governance and oversight exposes organizations to functional, security, and legal risks like hallucinated code, code leakage, and IP issues.



Executive Summary

How can organizations harness the full potential of generative AI for software engineering?

- Select and prioritize high benefit use cases.
- Mitigate risks around security, IP/copyright issues, and code leakage using a thorough risk management approach.
- Transform your software organization to ensure optimal usage of generative AI:
 - Augment your software teams with a generative AI assistant. A majority of junior (53%) as well as senior professionals (58%) believe that generative AI tools will augment their day-to-day work within the next two years. For instance, generative AI tools can help junior professionals learn faster and come up to speed quickly, while they allow senior professionals to focus on grooming juniors by ensuring their learning and retention, solving complex issues, and collaborating with business.
- Identify requirements for new capabilities and source them.
- Prepare for generative AI use by delivering technology prerequisites:
 - Build a repository of platforms and tools for a seamless and augmented software engineering experience.
 - Privately and safely contextualize generative AI assistants with organization's own content.
- Adopt a measurement protocol for generative AI impact monitoring and use case prioritization.
- Put people at the heart of this transformation by creating a learning culture at your organization.
 - Provide upskilling and cross-skilling opportunities.
 - Address employees' work displacement concerns.

Who should read this report and why?

This report provides insights into the use of generative AI for software engineering and offers recommendations that will be useful to organizations across industries in harnessing

1000+

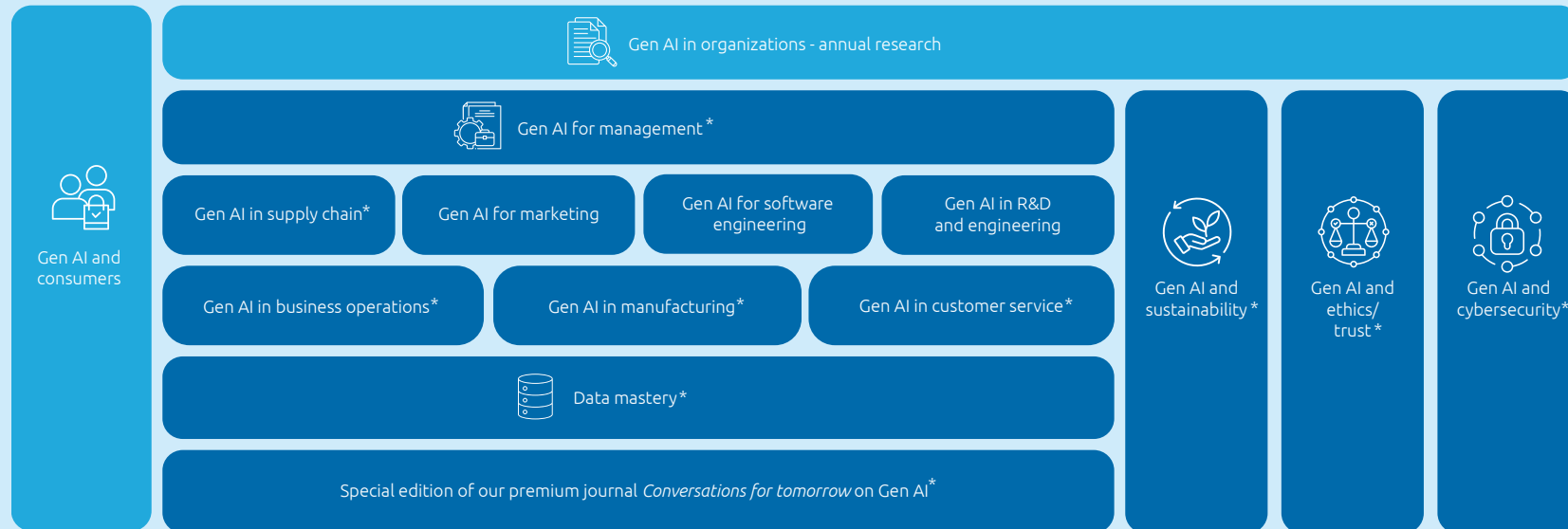
organizations with annual revenue greater than \$1 billion, represented by a minimum of one software professional and one software leader, are part of this research.

the full potential of generative AI for software engineering.

Business leaders in technology, IT, product, strategy, R&D/engineering, general management, and innovation who have responsibility for – and oversight of – their organization's software engineering function will find it particularly useful.

This report draws on insights from a comprehensive multi-sectoral survey of 1,098 senior executives (director level and above) and 1,092 software professionals (including architects, developers, testers, and project managers) from organizations with over \$1 billion in annual revenue. The report covers the major considerations for implementing generative AI in software engineering and includes in-depth qualitative insights from 20 industry leaders, professionals, and entrepreneurs.

This report is a part of Capgemini Research Institute's series on Generative AI



To find out more, please go to <https://www.capgemini.com/insights/research-institute/>

*Upcoming reports

Introduction

Since the dawn of the modern computer age, there has been a disconnect between natural language and machine language. With hardware and software advances, programming has evolved in waves over time and this gap has begun to close (see Figure 1).

This evolution now appears near complete, as natural language becomes the lingua franca. With recent rapid advances in AI and high-performance computing, we can now simply “chat” with computers and – through human supervision and accountability – let the AI assistant augment tasks ranging from programming, generating test cases and user stories, to documenting, among others. As Andrej Karpathy, one of the founders of OpenAI and former director of AI at Tesla, famously quipped following the introduction of ChatGPT: *“The hottest new programming language is English.”*³

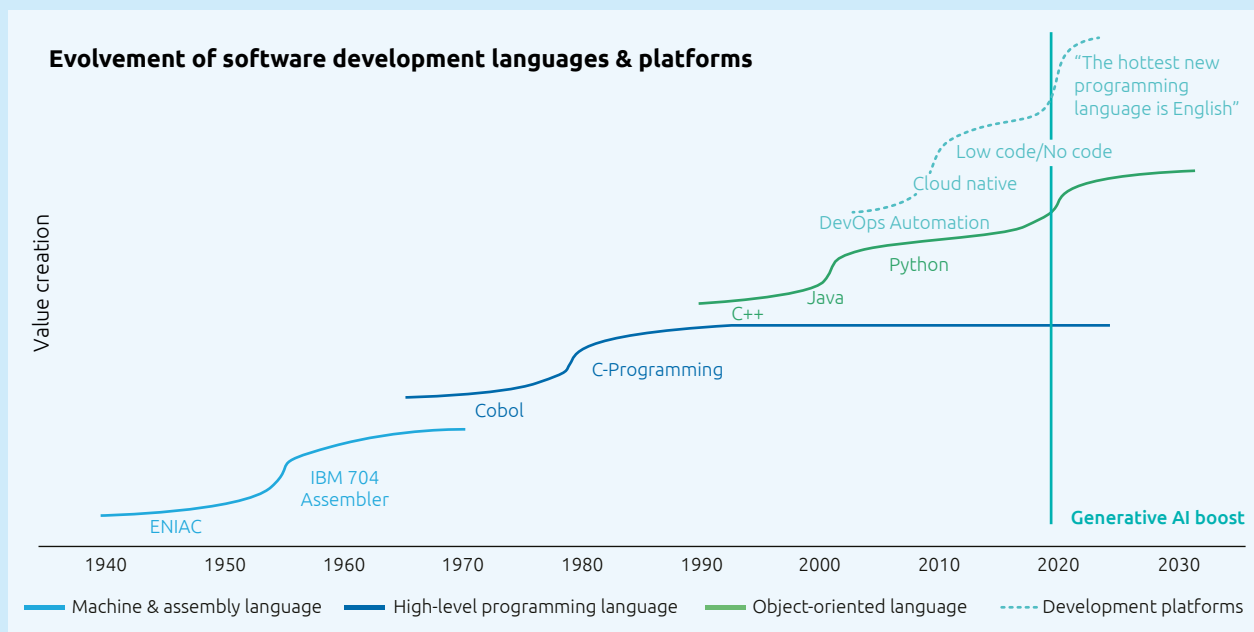
Today, by leveraging the power of large language models (LLMs), generative AI can enhance developers’ productivity, improve software quality, and accelerate time to market. Marco Argenti, Chief Information Officer at Goldman Sachs: *“Goldman Sachs is using artificial intelligence to turn software developers and others into superhumans.”*⁴

In generative AI, the software workforce has a tool to accelerate key tasks (such as design, coding, migrating, testing, deploying, support and maintenance) with minimal effort and a minimal learning curve.

Introduction

Figure 1.

Increasing levels of value creation from evolution of software development languages and platforms



Source: Capgemini analysis

Introduction

However, generative AI brings risks and challenges. Uncontrolled use can lead to hallucinated code, IP issues, private data leakages, and security vulnerabilities. Software engineering organizations need a new strategy and



implementation approach to harness the potential of generative AI while managing its risks. With this research we attempt to assess the impact of generative AI on the software engineering function, covering such questions as:

- How will generative AI impact the various stages of software development lifecycle (SDLC)?
- How can organizations quickly adopt and scale generative AI to drive productivity and innovation?
- How will generative AI impact software engineers' ways of working?
- What are the challenges for software engineering and how best can we manage the risks associated with generative AI?
- How can organizations continuously measure and optimize impact of generative AI on their software engineering function?

What do we mean by “Generative AI for software engineering”?

Defining the term “software”

Software is a strategic capability, transforming the way businesses design their products and services, function overall, compete, and provide value to customers. Software is vital to modern business, whether as a product itself or integrated into enterprise apps or products.

There are three main categories of software:

- **Business software:** Used by organizations to run, scale, and optimize day-to-day business functions and processes and/or interact with their customers and partners.

There are two broad types of business software:

- **Packaged software:** Third-party standard programs grouped to provide different tools from the same family in a package, commercially available under the licensor’s standard terms, payable with either a one-off or annual fee.

- **Custom software:** Specific, advanced programs developed for a specific purpose for an individual or company, which can be modified or changed. Custom software is not commercially available but is built and operated for internal purposes.
- **Consumer software:** Sold directly to end users, consumer software includes apps, web portals, and information tools such as maps, financial data, news, games, and music players.
- **Embedded software:** A piece of software to program hardware or non-PC devices to facilitate functioning. These are specialized environments and applications for a specific hardware stack with performance, power, and functionality requirement and constraints.

Generative AI has potential for all categories of software, but this research focuses largely on software engineering for custom, embedded, or consumer software which goes through the entire software development lifecycle.

Generative AI's potential for software engineering

Software engineering has shifted strongly towards greater automation and simplification, particularly with the advent of generative artificial intelligence (generative AI). The rise of large language models (LLMs) has been key. LLMs are deep-learning AI algorithms that can recognize, summarize, translate, predict, and generate content by building on very large datasets. They have facilitated the increasing adoption by consumers and organizations of software engineering.

Generative AI has the potential to transform the software engineering process, as it can be integrated into tech stacks to unlock new features and updates for software currently in use. Many leaders are striving to integrate AI-enabled plug-ins or incorporate AI-powered technology into their own enterprise and software engineering platforms. Our previous research shows that generative AI will assist in writing one out of every five lines of code in the coming year.⁵

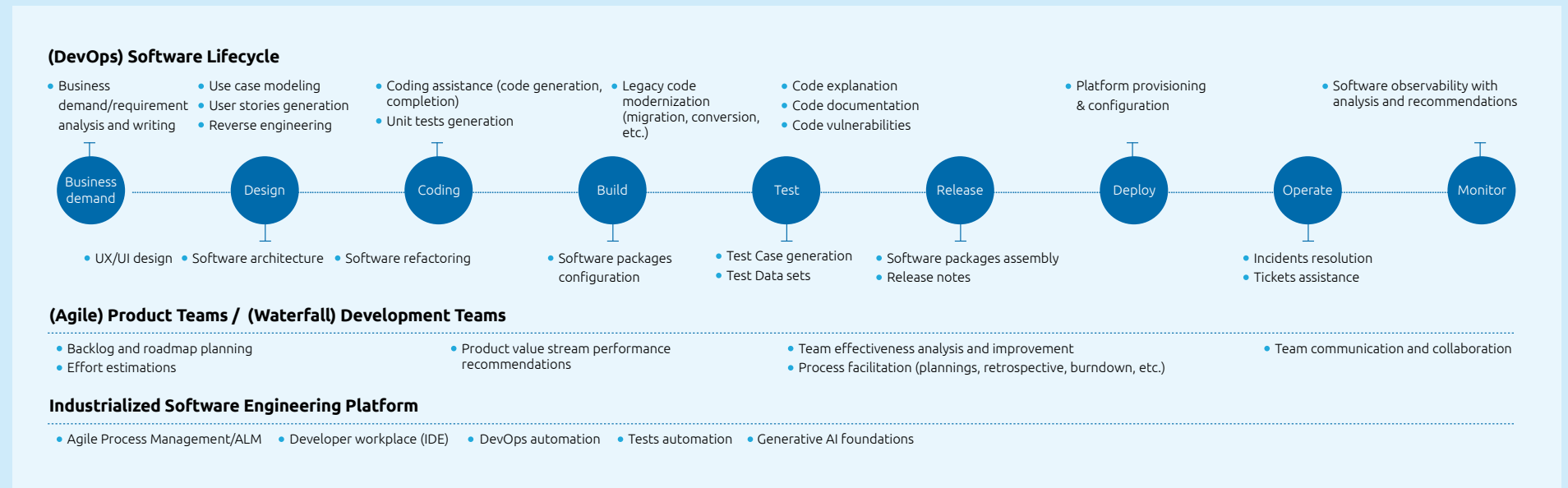
Generative AI's impact on the SDLC

With the increasing proliferation of software in products, services, operations; software teams are under pressure to deliver more, better, faster. Generative AI has the potential to yield benefits across the SDLC. Figure 2 shows some of the tasks and activities in SDLC that can benefit from the use of generative AI tools. It is worth noting that it is a subset of all activities encompassing SDLC. It can be integrated at any stage – from business needs analysis and writing agile user stories to software design, coding, documentation, packaging, deployment, testing, and operations – augmenting the work of software engineers and helping increase efficiency, improve quality, and enhance job satisfaction.

Generative AI also touches the roles of many data analysts, business analysts, platform/software designers, and software engineers, developers, and tester.

Figure 2.

Potential application areas of generative AI in the SDLC



Source: Capgemini Research Institute analysis.



01

Organizations are reaping significant benefits from leveraging generative AI for software engineering.

Figure 3.

One in two organizations adopting generative AI sees improvements in enabling innovative work and quality of software.

Percentage of organizations seeing benefits through the adoption of generative AI, as mentioned by software leaders



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executive Survey, April 2024, n = 412 software leaders that have scaled up or are running pilots with generative AI in software engineering.

Augmenting innovation and improving software quality are the leading benefits.

Three in five organizations see innovative work – for example, developing new features and services using software – as the biggest benefit of generative AI use in software engineering (see Figure 3). Of software professionals surveyed, 80% believe that, by automating simpler repetitive tasks, generative AI will free up time for them to focus on innovation and value-adding tasks, fostering greater creativity.

Akram Sheriff, Senior Software Engineering Leader (Gen AI, AI, ML) at Cisco Systems elaborates: *“One of the biggest drivers of generative AI adoption is innovation. Not just on the product side but also on the process side. While senior professionals are leveraging generative AI combined with their domain expertise for product innovation, junior professionals see value in AI process and tool innovation, and in automation and productivity optimization.”*

Generative AI also enables improvements in software quality. It can help deliver higher-quality code with fewer errors and improvements in test coverage and quality. Both factors give organizations a productivity boost at team and organizational levels. For example, Emirates NBD, a large banking group in the Middle East, not only accelerated developer productivity by up to 20% in complex tasks, but also improved the company's code quality by 20% by using GitHub Copilot's code suggestions.⁶

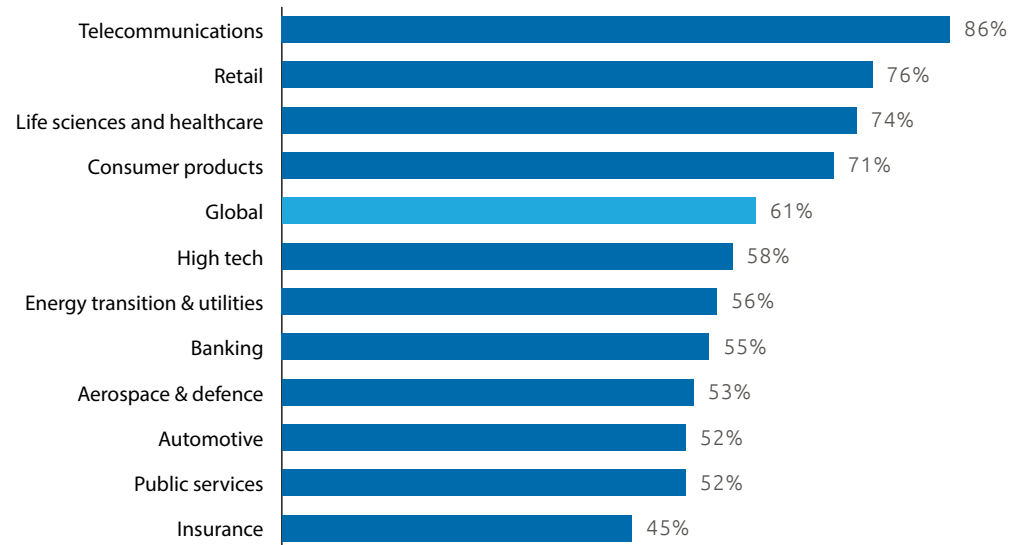
Head of AI at a leading Australian telco, explains: *"With use of generative AI for software engineering, the number of test cases could be increased by 30%, greatly enhancing test coverage and quality."*



Figure 4.

Telecom and retail sectors see enablement of innovative work as a top benefit from generative AI.

Percentage of organizations by sector, who have active initiatives and see enablement of innovative work as a top benefit, as per software leaders



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executive Survey, April 2024, n = 412 senior executives that have scaled up or running pilots with generative AI in software engineering.

For telecom businesses, generative AI can play a significant role in the development of such data-powered, innovative applications as network management and maintenance as well as customer service/sales apps offering hyper-personalization. BT Group's Digital unit has an AI-powered product lifecycle management strategy. Within four months of deploying Amazon's CodeWhisperer, it had automated nearly 12% of repetitive work, allowing the pilot workforce to focus on more strategic goals.⁷

Similarly, the retail industry is leveraging generative AI to gather and analyze customer preferences, competitor insights, past sales history, etc., and create robust and precise requirements documentation as the basis of engaging customer-facing apps. Wayfair, a home goods company, is considering using generative AI to reduce the technical debt accumulated in their software stack over years.⁸

Organizations with active generative AI initiatives have seen an average 7–18% improvement in productivity across the SDLC.

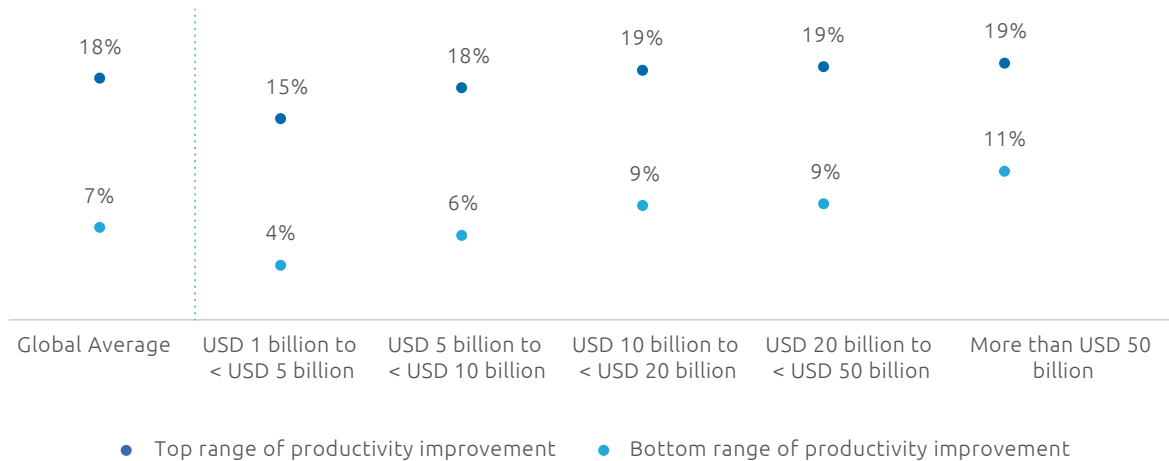
Those organizations actively using generative AI in software engineering have seen an average total productivity improvement of 7–18% across the SDLC today, compared to non-usage of generative AI. The increasing maturity of tools and processes along with growing professional experience, means productivity is likely to continue to improve.

We also found that productivity advantage increases with organization size (see Figure 5).

Figure 5.

Larger organizations have seen greater productivity improvement with generative AI.

Productivity improvement range of a software professional, grouped by organization revenue size



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executive Survey, April 2024, n = 412 software leaders that have scaled up or running pilots with generative AI in software engineering. Top and bottom productivity ranges are found by the 80th and 20th percentile respectively of individual productivity improvement data.

Ancileo, Singapore-based insurance software-as-a-service (SaaS) company, used generative AI to increase developer productivity. Sylvain Dutzer, Chief Technical Officer, Ancileo: *"Ancileo is using Amazon Q to supercharge our developers by helping them understand existing codebase and troubleshoot directly in their integrated development environment (IDE). This allows our team to reduce time resolving coding-related issues by 30%. Even our architects use it to help find the best solutions to specific problems based on context."*⁹

Nitin Tandon, Chief Information Officer of financial services firm Vanguard: *"We are enabling productivity gains for developers by experimenting 'rapidly and safely' with generative AI tools — with human oversight and expertise."*¹⁰

Improvements in coding speed (78%) and testing speed (54%) are the top reasons cited for this improvement. Generative AI can produce test cases directly from requirements, with significant time savings. Where testing an app requires certain application programming interfaces (APIs), AI test code generators can create these snippets quickly.

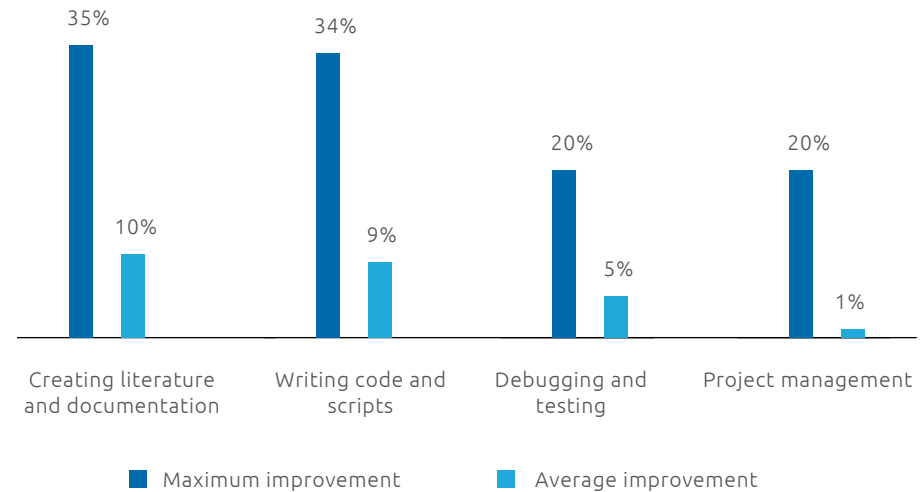
Generative AI yields productivity improvement for a set of specialized software tasks

We analyzed specific tasks from a software professional's daily work to understand the impact of generative AI. Some tasks are better suited for generative AI, given the maturity of the tools available and the experience of the workforce. As shown in Figure 6, creating literature and documentation, and writing code and scripts show the greatest timesaving. This tapers off for the remaining major task categories in the SDLC. However, as toolchains and platforms improve, this benefit is likely to spread. It is important to note that saving time using generative AI tools is significantly different from saving cost. Assessing cost savings was not a part of the scope of this research.

Figure 6.

Generative AI shows significant productivity improvement in terms of time savings for documentation and coding assistance tasks.

Maximum and average time savings through generative AI usage for a set of activities (current estimates)



Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professionals Survey, April 2024, n = 368 software professionals that are actively using generative AI. Maximum improvement is represented by the 95th percentile's results, while an average user is represented by the statistical average.

Organizations are utilizing productivity gains on innovative work and upskilling, not headcount reduction.

According to our survey, 79% believe generative AI will significantly reduce the workload and free up additional time for software professionals. This freed up time is being used for higher-value-adding tasks including enhanced innovation and upskilling, as shown in Figure 7.

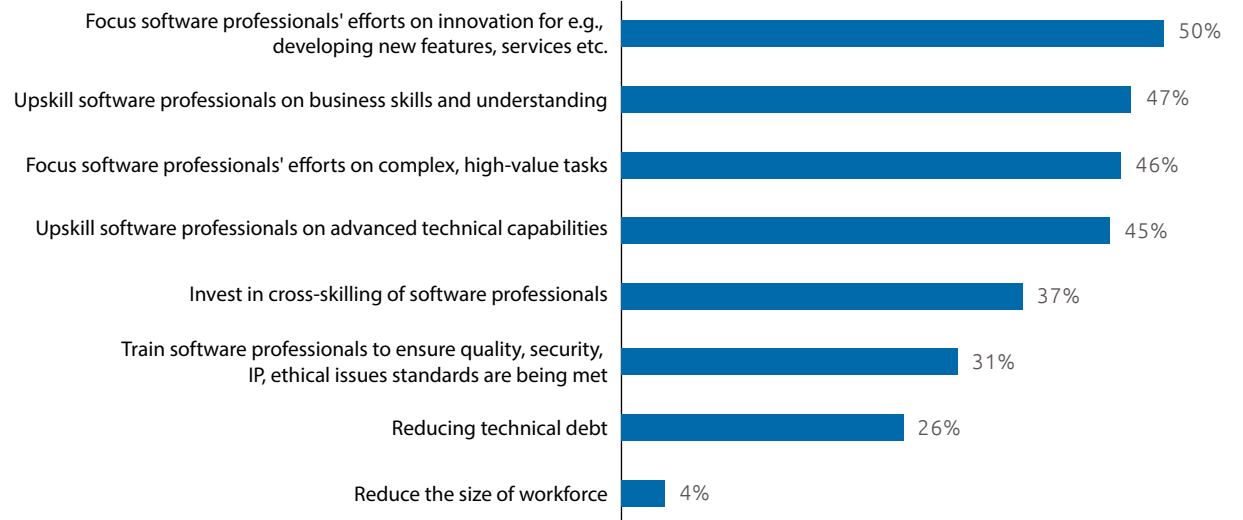
Mousumi Bhattacharya, Director of IT at Centene, a US-based managed care company: *“Generative AI has tremendous potential to improve productivity by shifting professional efforts and time from mundane and repetitive things to more meaningful, creative and challenging tasks.”*

Stephane Dupont, EVP and Head of Operations at Airbus, the leading European aerospace company: *“I see it as a coding assistant, giving developers more time to think about the architecture, the new features, next steps, quality, etc., and spending less time on pure code development.”*

Figure 7.

Innovative work and upskilling are the top areas where organizations are channelizing productivity gains.

How is your organization planning to leverage the additional time freed up by generative AI?



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executives Survey, April 2024, N = 870 senior executives who believe that generative AI will free up additional time for software professionals

Brian Lanehart, president, and CTO of financial technology provider Momnt: *“Being able to completely communicate an entire application request to generative AI will reduce a task timeline significantly. That means an engineer or team is freed up to think creatively or strategically.”*¹¹

Reducing headcount is the least-adopted route (taken by only 4% of responding organizations); and new roles, such as generative AI developer, generative AI Architect, AI platform architect, prompt engineer, etc. have evolved. The head of AI at a leading Australian telco: *“Even as autonomous vehicles are a reality, human supervision and ability to take control is still required. Similarly, software engineers won't be replaced by generative AI – they will start thinking about the actual design process, long-term strategy, next phase of software, etc. rather than spending a year writing code.”*



“I see it as a coding assistant, giving developers more time to think about the architecture, the new features, next steps, quality, etc., and spending less time on pure code development.”

Stephane Dupont

EVP and Head of Operations at Airbus,
the leading European aerospace company

Generative AI benefits in software engineering extend to job satisfaction and happiness.

Our research shows that generative AI has a positive impact on software professionals' job satisfaction and reduces attrition rates (see Figure 8). Fabio Veronese, Head of ICT Industrial Delivery at ENEL Group: *"We are more ambitious. For us, improving development productivity with generative AI is not just about lines of code. It is also about developer experience."*

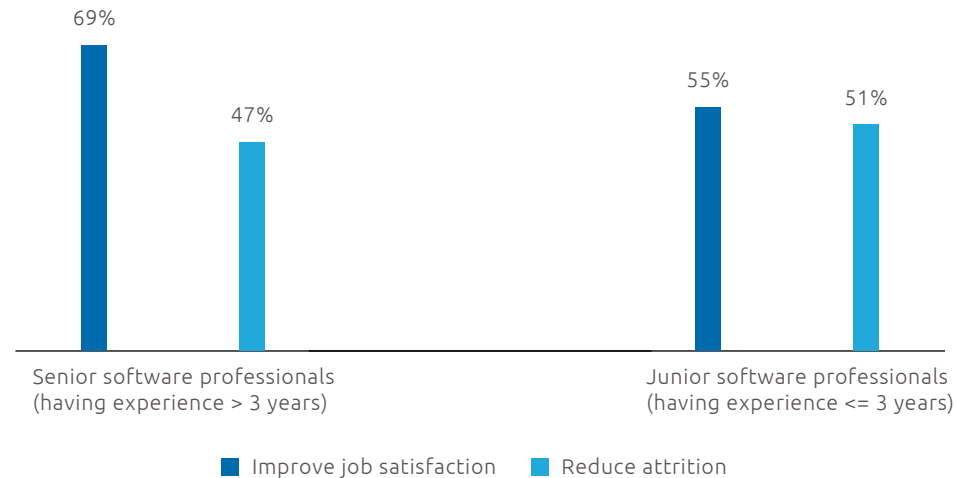
69%

Senior software professionals believe that generative AI will have a positive impact on job satisfaction

Figure 8.

Senior and junior professionals see positive impacts of generative AI on their job satisfaction

Extent to which generative AI impacts these areas within your organization's software engineering function



Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professionals Survey, April 2024, N = 215 software professionals

Most of the current workforce sees generative AI as a strong enabler and motivator – 35% associate it with being “assisted and augmented,” and 24% feel “excited and happy” about its adoption (see Figure 9).

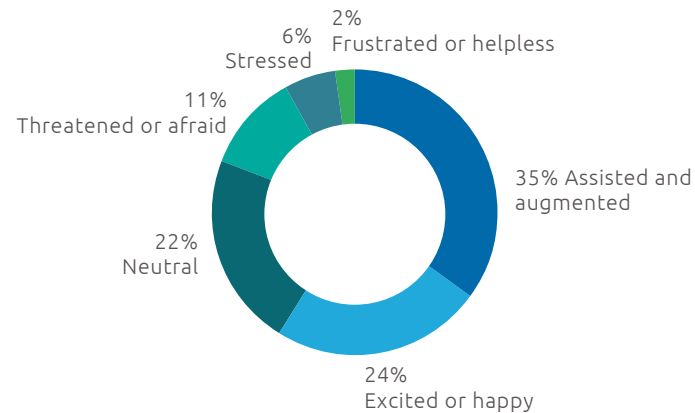
While there is currently an emphasis on generative AI’s utility in code completion and writing, three in four senior executives believe it will significantly transform their software engineering organization.

Tommy MacWilliam, Engineering Manager for Infrastructure at Figma: *“Personalized, natural language recommendations are at the fingertips of all our developers. Our engineers are coding faster, collaborating more effectively, and building better outcomes.”*¹²

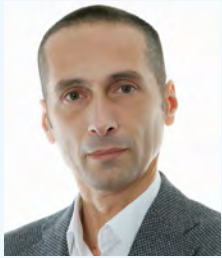
Figure 9.

Most of the workforce feels positive about generative AI tools for software engineering.

How does the workforce feel as regards to the adoption of generative AI



Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professionals Survey, April 2024, N = 1,092 software professionals



“We are more ambitious. For us, improving development productivity with generative AI is not just about lines of code. It is also about developer experience.”

Fabio Veronese

Head of ICT Industrial Delivery at Enel Grids



02

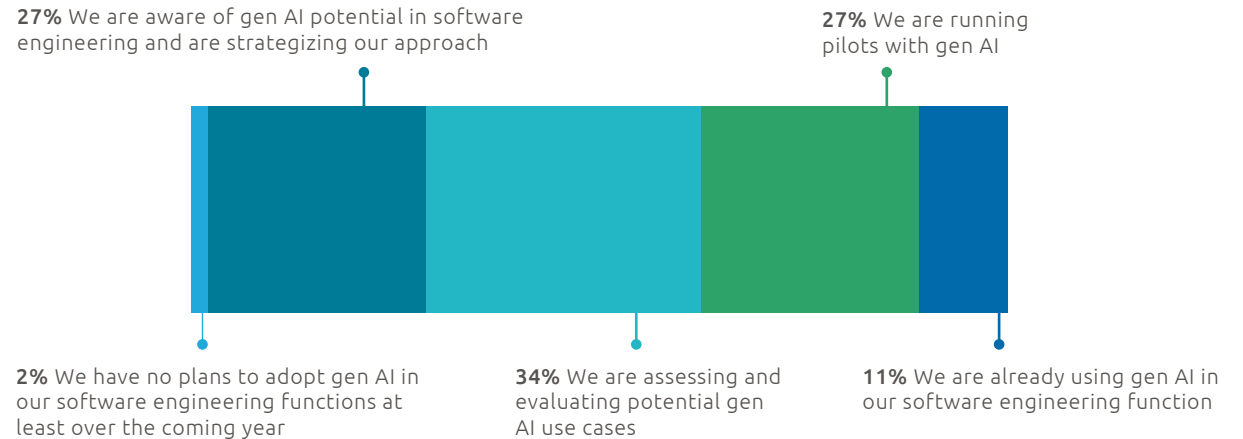
Generative AI adoption is at an early stage but will accelerate sharply

Generative AI adoption in software engineering is at an early stage, with larger organizations showing higher rates of adoption.

Currently, 27% of organizations are running generative AI pilots, with 11% leveraging generative AI for software development tasks (see Figure 10). The remaining majority are at an earlier stage.

Figure 10.

Only one in ten organizations currently uses generative AI for software engineering.



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executives Survey, April 2024, N=1,098 senior executives

Adoption rates show a strong correlation with revenue size (see Figure 11). Cost of tools, lack of upskilling and training budget, privacy and security concerns, and the high costs of safeguarding against them¹³ are some of the common deterrents for smaller-sized organizations.

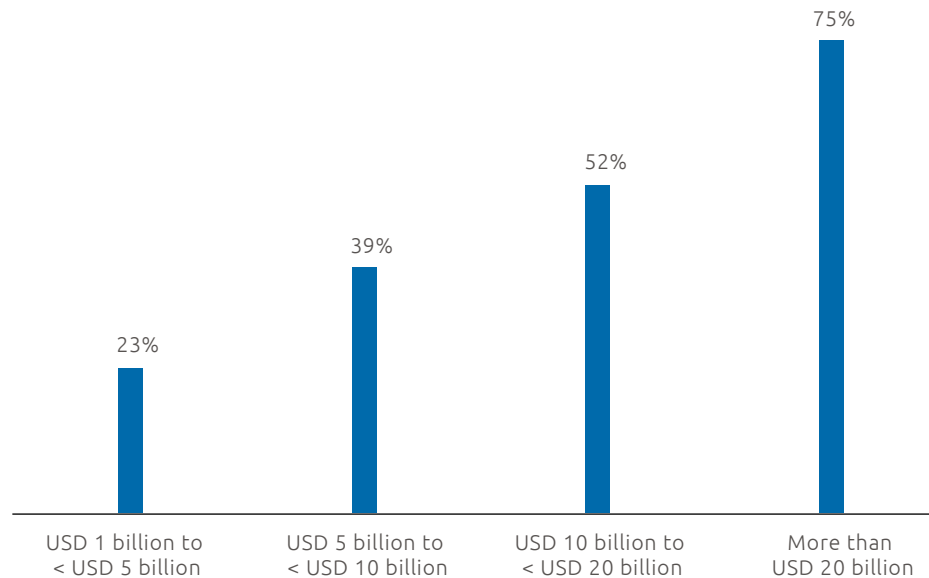
75%

Organizations with more than USD 20 billion of annual revenue have adopted (piloted or scaled) generative AI for software engineering.

Figure 11.

Larger organizations are further along their adoption journey than their smaller counterparts.

Percentage of organizations that have adopted (piloted/scaled) generative AI, grouped by revenue size



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executives Survey, April 2024, N = 1,098 senior executives.

Adoption is expected to accelerate – more than four in five software professionals will leverage generative AI tools and solutions by 2026

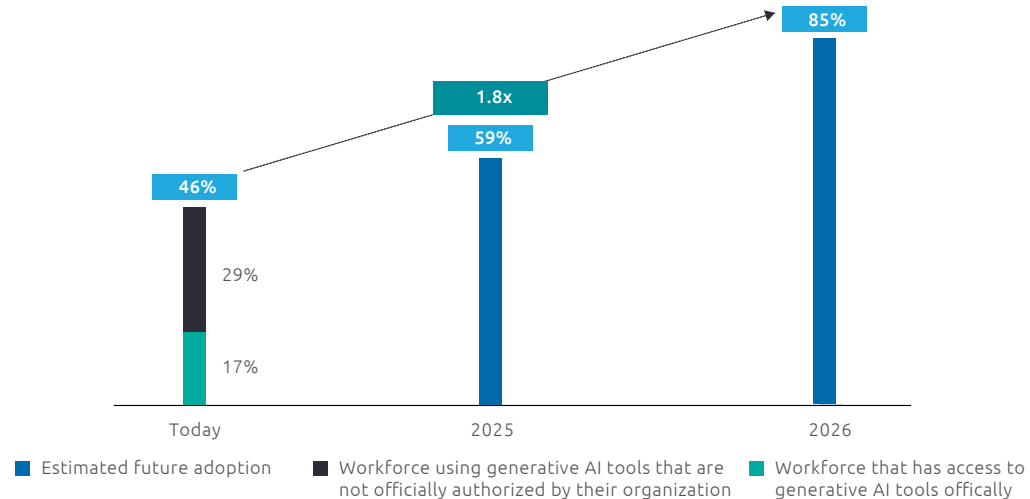
As organizations unravel the use cases of generative AI in software engineering and start realizing benefits, adoption is predicted to accelerate (see Figure 12). Nearly half of the workforce (46%) today is making use of generative AI tools for software engineering in any type of use – training, experimenting, piloting, or implementing in real environments. It is expected to nearly double in the next two years.

Business users will play a role in generative AI's growth in software engineering. Director at a leading biopharma company: *"Not everyone is going to be an app developer. But generative AI will unlock the capabilities of business users to some extent and make them more independent, allowing them to self-create some code or apps as needed."*

Figure 12.

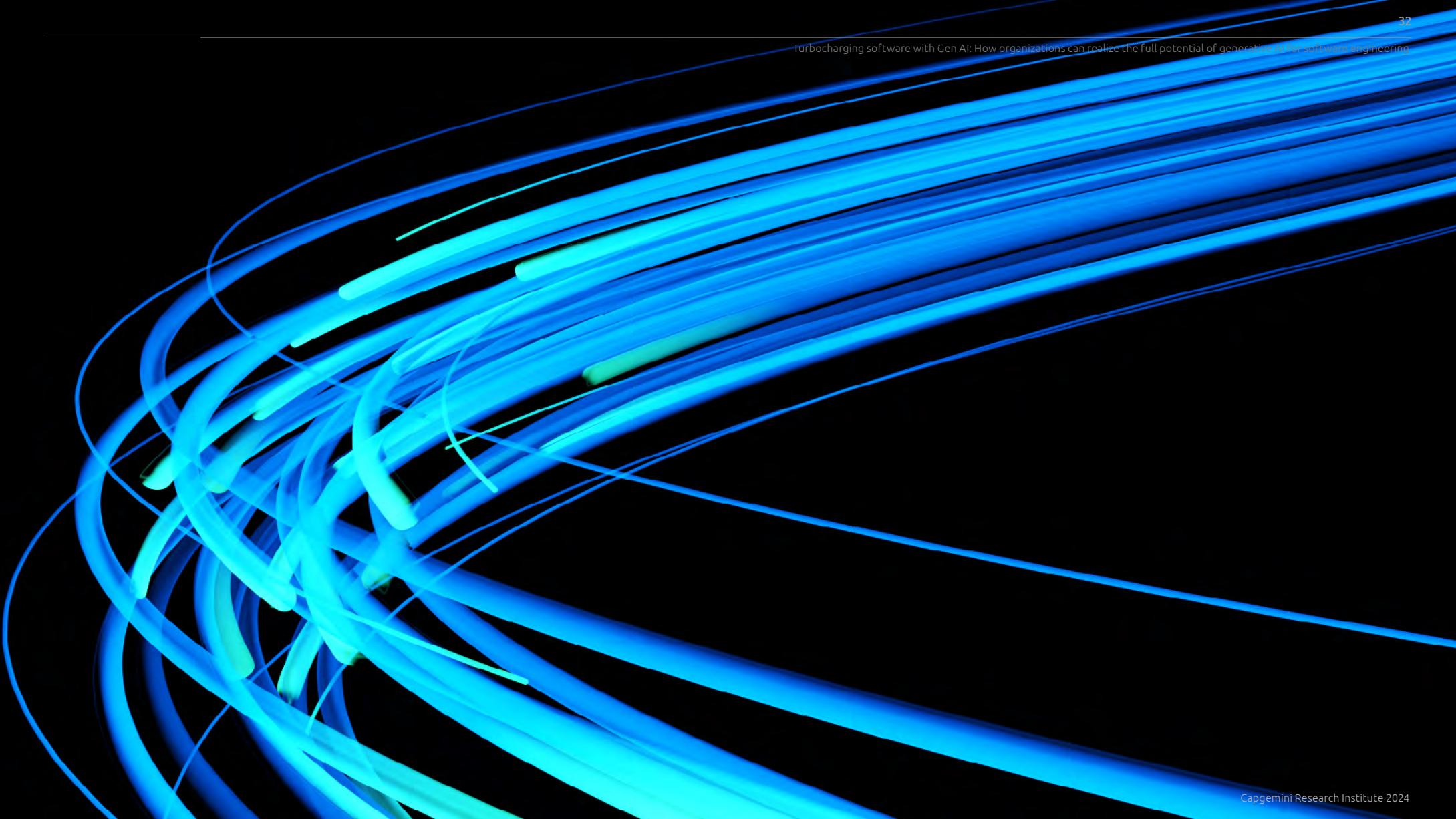
It is estimated that more than four in five software professionals will leverage generative AI tools and solutions by 2026 – both officially and with unauthorized access

Percentage of workforce leveraging generative AI tools in the workforce



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executives Survey, April 2024, N = 1,098 senior executives.

Note: Today's data is representative of the survey results, while 2025 and 2026 is estimated on senior executive's future deployment plans and the trends in unofficial usage



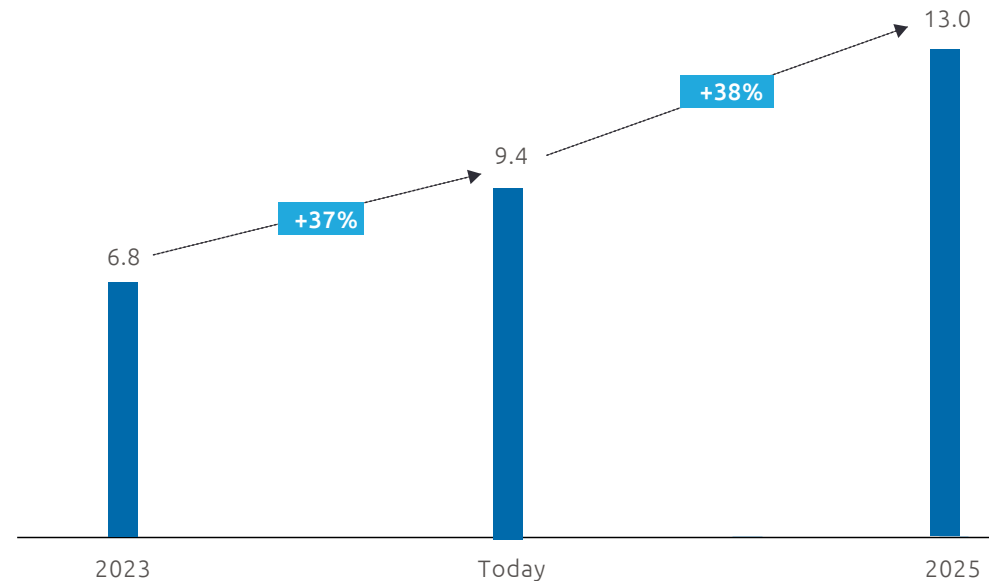
As we will discuss further, a considerable segment of the software workforce (63% of those using generative AI) currently leverages generative AI unofficially (see Chapter 3).

Organizations overall are increasing their spending to keep up with the growing demand for generative AI (see Figure 13).

Figure 13.

Organizations are increasing their investment in generative AI for software engineering.

Average yearly investment in gen AI for software engineering (USD Million)



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executives Survey, April 2024, N = 1,098 senior executives

Coding assistance is the leading use case, but generative AI is being used in other SDLC activities.

Applications of generative AI in software engineering go beyond coding assistance, ranging from test case generation to business requirements/demand analysis and writing to legacy code modernization. Of our respondents, 70% agree that generative AI's potential extends beyond writing code.

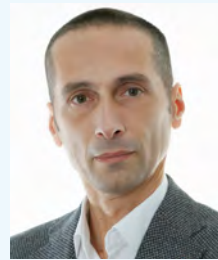
We list below some interesting uses cases and applications:

- Coding assistance:** This is the most widely recognized generative AI use case due to the significant impact it can have on productivity and accuracy. The developer can share target functionality, and the tool uses LLMs to generate the code. Coding assistants such as Microsoft Github Copilot, AWS Amazon Q Developer, Google Gemini Code Assist, and Codeium can generate code from natural language and assist in syntax completion, creation of code boilerplates and prototypes, etc. Based on past data, algorithms can detect source code patterns that are likely to introduce bugs and suggest code changes to help address this.¹⁴
- UX design assistance:** Generative AI allows developers to test UX designs better and faster. It can capture user inputs, provide recommendations, design customized and engaging user experiences, generate novel design elements, and automate the creation of prototypes to some extent. Yannis Paniaras, Principal Designer, Microsoft Digital Studio: *"The AI transforms into the conductor of the user experience. This enables our designers to move away from defining fixed flows and embrace non-deterministic design orchestrated by the AI."*¹⁵
- Business requirements/demand analysis and writing:** In addition to taking natural language inputs from analysts, generative AI can analyze large amounts of data, including customer reviews, market research, and industry best practices, to identify user needs and preferences and translate them into functional and system requirements quickly. It can help to track changing requirements and improvise and validate requirements documentation by analyzing it for completeness and ambiguity, providing feedback and suggestions. It can also assist in generating epics and stories, contextualized with organization's private documentation corpus.

While this is not one of the top use cases being adopted (as we will see in the next sub-section on use case adoption) the availability of tools and industry examples indicate the potential of using generative AI tools for this use case.

For instance, Siemens equipped field and shopfloor workers to document and report issues in real-time using natural speech which leverages AI to analyze it and routes it to the appropriate design, engineering, or manufacturing experts—in the specific language they require.¹⁶ POPal is a user story and Jira issue generator leveraging generative AI to create and refine user stories with title, benefit hypotheses, and acceptance criteria.¹⁷ A Senior Architect at a Fortune 500 industrial conglomerate: *"We are piloting with generative AI to convert whole meetings into user (JIRA) stories with requirements and acceptance criteria and push them into our requirements tool. The added benefit is that when we give requirements for phase two, AI will have the requirements for phase one and can craft/add to them accordingly."*

- **Legacy code modernization:** Generative AI can be a game changer for legacy migration projects, significantly accelerating code migration and reducing downtime. It can automatically convert legacy code to natural language, making it simpler to understand the complex functionalities and business logic embedded in the code. Many legacy applications and systems run on decades-old code, and when the current workforce cannot understand or enhance the codebase, this leads to a pile-up of technical debt. Generative AI helps organizations start afresh. It can also help document current architecture and optimize target architecture by recommending optimized partitioning, refactoring codebases for cloud, migrating data to cloud, and automating deployment.¹⁸
- **Testing:** In software testing, generative AI accelerates quality engineering and enhances test coverage (the share of code that's tested) and overall testing efficiency. Generative AI quickly understands user stories and requirements, almost instantly applying proven test design methodologies. It generates comprehensive test cases much faster than manual methods and reduces human error, ensuring thorough and precise testing. Utilizing various testing techniques, such as boundary value analysis, state transition testing, and equivalence partitioning, AI ensures all scenarios, including edge cases, are comprehensively covered. It adapts to specific project needs, generating highly relevant test cases and handling



“Enel Grids uses generative AI to create user acceptance tests and specifically to explain test scenarios to businesspeople whose IT knowledge might be limited.”

Fabio Veronese

Head of ICT Industrial Delivery at Enel Grids

projects of any size, from small to large-scale applications. Over time, it improves by refining outputs based on feedback and new data, quickly updating test cases to reflect changes in the application. Additionally, generative AI can create a baseline of automation scripts, ensuring a strong foundation for automated testing processes. Fabio Veronese, Head of ICT Industrial Delivery at Enel Grids shares, *“Enel Grids uses generative AI to create user acceptance tests and specifically to explain test scenarios to businesspeople whose IT knowledge might be limited.”*

- **Deployment and maintenance:** Generative AI can assist in activities in the deployment and rollout phase, including infrastructure configuration, containerization, and orchestration tasks as well as such maintenance activities as incident and log analysis¹⁹ (see Figure 15). Director Product Marketing at a leading technology major shared, *“Before a product rollout, our company uses generative AI to assist in the detection of any vulnerabilities that might be there in the code. As a result, what used to take 24 hours earlier, is now accomplished in just one hour!”*
- **Reduced technical debt:** Generative AI can help control growth of technical debt by generating optimized code/output and by assisting the clean-up and migration of legacy applications. Wayfair relies on generative AI to reduce technical debt accumulated while trying to fix

technology problems quickly. This includes legacy code in PHP and old database code in SQL, as well as code written by developers who have left the company.²⁰

Enhanced collaboration: Enabling seamless communication between team members, as well between technology and business teams, whether communicating requirements or explaining what the code does in natural language. 78% of software professionals are optimistic about generative AI’s potential to enhance collaboration between business and technology teams. Avanthika Ramesh, Director of Product Management at Salesforce, the leading CRM software company: *“Generative AI can promote collaboration and cross-functional thinking amongst software engineers. AI can help them understand how their solution fits into the broader solution and how to integrate their code”.*

Generative AI can go beyond coding assistance for developers. Established tools like Microsoft GitHub Copilot²¹, Google Gemini Code Assist²², and Amazon Q Developer²³ are catering to quite a few use cases in the SDLC value chain like optimizing code, writing test cases, checking for vulnerabilities, modernize legacy code, etc. e.g.,

- **Microsoft** is developing the GitHub suite beyond Copilot, with Copilot Enterprise and GitHub Copilot Workspaces that caters to multiple use cases across the SDLC.



“Generative AI can promote collaboration and cross-functional thinking amongst software engineers. AI can help them understand how their solution fits into the broader solution and how to integrate their code.”

Avanthika Ramesh

Director of Product Management at Salesforce

Microsoft Copilot Studio has usage around documentation and business analysis, benefiting from “low code” contextualization using organization’s own content. Microsoft’s Sketch2Code has applications in design and modelling as it can translate handwritten user interface design to valid HTML markup code.²⁴

- **AWS** is extending its Amazon Q suite through Amazon Q Developer not only for coding assistance but also assisting in code modernization, changing application architecture, front end development, among many other use cases. In addition to existing AWS Bedrock services which addressing other SDLC stages through custom solutions.
- **Google** is progressively improving upon Gemini Code Assist for rich coding use cases including assisting in generating and debugging code, generating unit tests, troubleshooting code with issues, offering suggestions for optimization, improving code readability, etc. in the existing GCP Vertex services which are addressing other SDLC activities through custom solutions.
- Additionally, software engineering focussed solutions like Atlassian, GitLab are continuously embedding generative AI in their respective solutions (Atlassian’s ALM, Gitlab’ DevSecOps) for coding assistance, code modernization, etc.

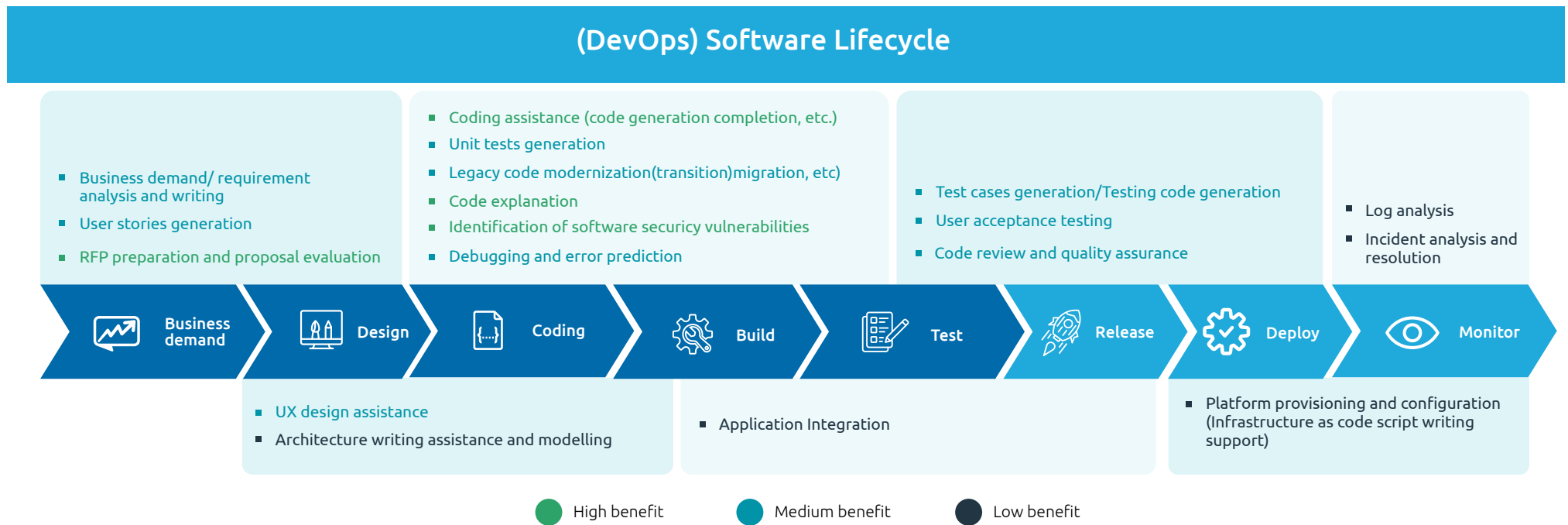
In addition, as illustrated in the table below, there are many upcoming specialized tools from startups that cater to specific use cases:

| SDLC phase | Illustrative list of startup solutions |
|----------------------------|--|
| Business Demand Analysis | <ul style="list-style-type: none"> ▪ 8base's Archie²⁵ for ideation, blueprint generation, detailed requirements writing in addition to architecture design support. ▪ GPT powered tools like PRD wizard, and WritemyPRD²⁶, etc. for product requirements documentation |
| Design | <ul style="list-style-type: none"> ▪ Eraser's DiagramGPT to create data flow diagrams, architecture diagrams, etc.²⁷ ▪ Mintlify to generate code documentation²⁸ |
| Testing | <ul style="list-style-type: none"> ▪ Diffblue for writing test cases.²⁹ ▪ Deepcode's Synk for securing code³⁰ ▪ Sapient for generating test cases.³¹ |
| Deployment and maintenance | <ul style="list-style-type: none"> ▪ Grit for generating release notes by analyzing commits, issues, and differences; dependency upgrades |
| Legacy code modernization | <ul style="list-style-type: none"> ▪ Bloop AI to help teams modernize, and understand their legacy code³² |
| DevOps | <ul style="list-style-type: none"> ▪ OpenText™ DevOps Aviator for faster application delivery³³ |

As Figure 14 shows, our survey highlights the areas where generative AI can deliver a high, medium, and low levels of benefit.

Figure 14.

Generative AI can deliver a range of benefits throughout the SDLC.



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executive and Software Professional Survey, April 2024, N = 1,092 organizations

Note: This chart shows a selection of use cases that we evaluated in greater detail in the research. Figure 2 gives a longer list of all potential use cases for generative AI in software engineering. Both these figures represent a subset of all SDLC activities.

Most use cases are in early stages of adoption.

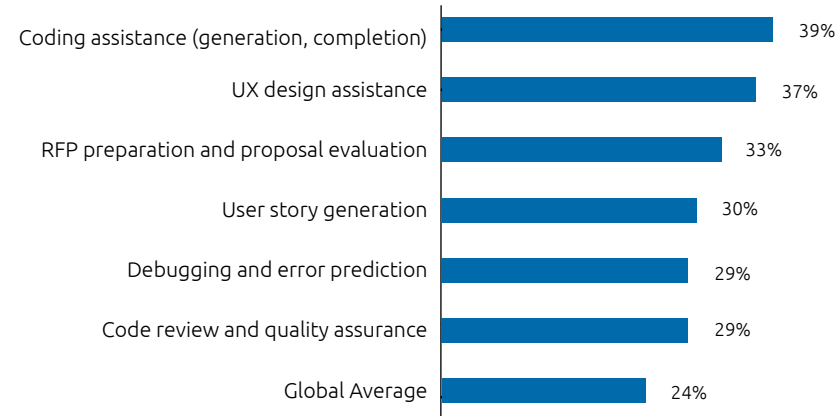
The adoption of most use cases is at a nascent stage, and few organizations are implementing or piloting them. Coding assistance, the highest adopted use case, is at 39%, while test case generation stands at 26% (see Figure 15). Globally less than one in four organizations are focusing on any use case on average.

This may be attributable to the relatively recent rise of generative AI technology, people mindset, organizations being slow to provide the necessary tools, training and governance, and poor selection of use cases (more on this in the final chapter).

Figure 15.

Top generative AI use cases for software engineering by rate of adoption

Percentage of organizations focusing on the following use cases for implementation (pilot or scaled)



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executive Survey, April 2024, N = 1,098 senior executives; Software Professionals Survey, April 2024, N = 1,092 software professionals.

Note: Adoption and implementation refers to any scale - from trial/pilot, to a scaled implementation.

Coding assistance: 77% of organizations agree that using generative AI for code assistance brings significant benefits. Duolingo saw a 25% increase in developer speed using CoPilot, GitHub's tool. It increased developer productivity by limiting context switching, reducing the need to manually produce boilerplate code, helping developers focus on complex tasks.³⁴ Duolingo Chief Technology Officer Severin Hacker: *"A tool like GitHub Copilot is so impactful because suddenly engineers can make changes to other developers' code with little previous exposure."*

Avanthika Ramesh from Salesforce: *"Engineers have been saying that they don't need to search for certain code snippets as they're already on hand. They're pleasantly surprised about how intelligent AI is!"*

Test case generation: 61% of organizations agree that using generative AI for test case generation can give significant benefits. Goldman Sachs was able to increase legacy code test coverage from 36% to 72% in less than 10% of the time it would take to do so manually, thus ensuring higher application stability and faster integration. Average time to write each unit test came down from 30 minutes to 10 seconds.³⁵

Qualtrics, a Seattle-based company that builds experience management software, reported that unit tests, which previously took a full day to write, take 10 minutes with their generative AI tool.³⁶

Legacy code modernization: The main benefit of generative AI is that it makes such large and previously unthinkable migration projects feasible. Peter Schrammel, CTO and Cofounder of Diffblue, a leading generative AI startup for software engineering says: *"The big gain for application modernization and migration is being able to undertake such big projects. It gives you confidence in reaching the right test coverage at lower cost and faster, making it affordable for large enterprises and drastically reduces the time taken."*

Companies like ADP, UK-based payroll and HR systems provider, and JPMorgan Chase already use generative AI for system modernization and migration.³⁷

Nicole Onuta, Manager Artificial Intelligence Lab at a large financial services company in Netherlands says, *"I have seen a huge benefit in transitioning the code from a legacy programming language to a new one using generative AI."*³⁸ Similarly, tech company in the health and security sector, Leidos, has reduced by 50% the time spent dealing with legacy code using generative AI tools. Migrating a certain piece of code from Oracle to PostgreSQL which would have otherwise taken a full sprint could be 80–90% completed, with use of generative AI, in a matter of minutes.³⁹



"I have seen a huge benefit in transitioning the code from a legacy programming language to a new one using generative AI."

Nicole Onuta

Manager Artificial Intelligence Lab at a large financial services company in Netherlands

Generative AI will play a larger role in assisting the software workforce within the next two years

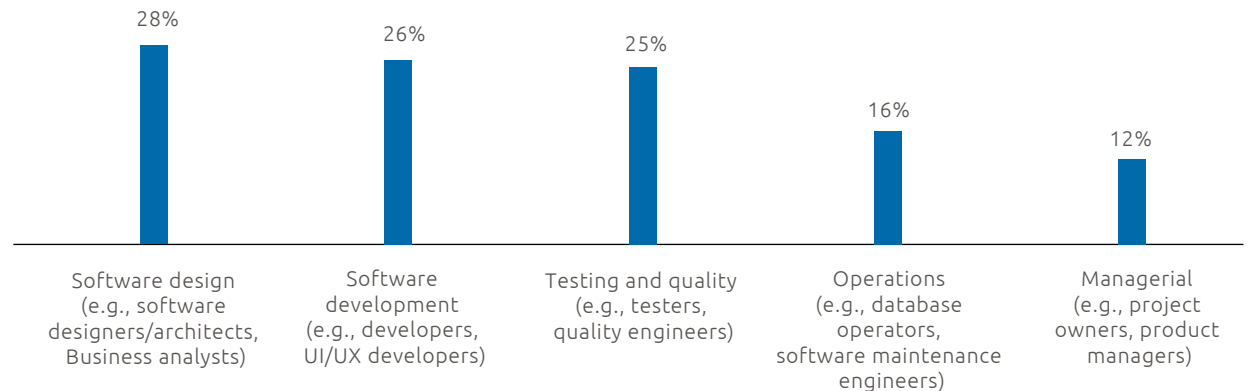
The software workforce is optimistic. They expect over a quarter of all work in software design, development, testing, and quality to be augmented and assisted by generative AI in two years (see Figure 16).

Senior director of software product engineering at a global pharmaceutical major: *“Currently, assisting in code generation and writing test cases are number one priority, but use cases like bug fixing and documentation are fast emerging, with others like UX design, requirement writing, etc. just around the corner.”*

Figure 16.

Generative AI has the potential to assist with more than a quarter of software design, development, and testing work in two years.

Percentage share of work that can be augmented or assisted by Generative AI by the next two years, grouped by role



Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professional Survey, April 2024, N = 1,092 software professionals.



“The big gain for application modernization and migration is being able to undertake such big projects. It gives you confidence in reaching the right test coverage at lower cost and faster, making it affordable for large enterprises and drastically reduces the time taken.”

Peter Schrammel

CTO and Co-founder of Diffblue

03

Lack of prerequisites and unofficial usage of generative AI pose significant functional, security, and legal risks

To fully leverage the potential of generative AI, software organizations need to provide fundamental prerequisites, such as platforms and tools and processes/workflows.

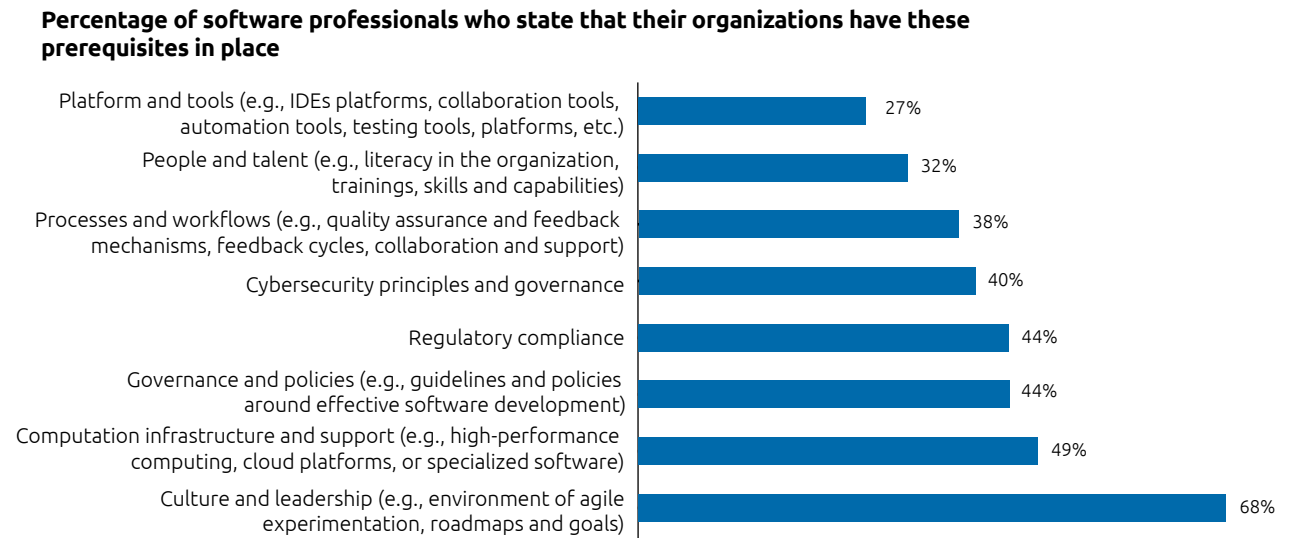
Less than a third of organizations have the platforms and tools or people and talent needed to implement generative AI for software engineering.

Only around a third of organizations are suitably equipped (see Figure 17). Most lack the prerequisites for implementing generative AI for software engineering – aside from culture and leadership, which two in three organizations (68%) claim is in place.

Senior director of software product engineering at a global pharmaceutical major: *“One challenge towards generative AI adoption is the lack of tools which can be completely integrated into the software workflow. In the initial days of adoption this was a bigger concern, but as generative AI is maturing, it’s not long before vendors will be able to mitigate this issue satisfactorily.”*

Figure 17.

Platforms and tools, along with talent, are the top prerequisites currently lacking for the implementation of generative AI.



Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professionals Survey, April 2024, N = 1,092 software professionals

Sustainability in generative AI for software is frequently overlooked.

Environmental sustainability is a key concern for organizations worldwide, but the impact of using generative AI in software engineering is underappreciated by most software professionals.

- 40% believe generative AI use in software engineering will increase the carbon footprint of their organization's software function, whereas 35% disagree – indicating confusion about the impact.
- Only 19% of software professionals rank carbon footprint and impact on sustainability as a top five challenges of implementing generative.
- Only 10% of organizations cover generative AI's carbon footprint in any reskilling program.
- 64% of organizations have no governance framework in place to cover sustainability and carbon footprint monitoring related to generative AI for software initiatives.



This indicates a need to bring sustainability into leadership discussions, define proper safeguards for generative AI use and consider sustainability when selecting tools and approaches.

More than three in five organizations lack a governance framework and upskilling program for generative AI.

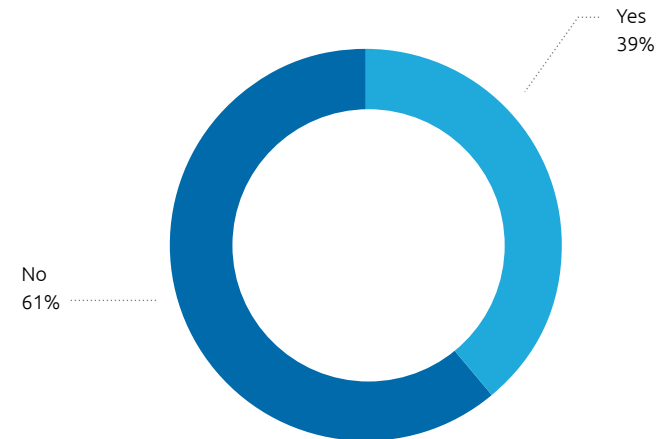
Most organizations (61%) lack a relevant governance framework and upskilling program (see Figure 18). A governance framework that defines standards and guidelines to ensure generative AI implementation aligns with organizational priorities and objectives is critical to success. Key aspects to address include code provenance, reducing bias, explainability of the model and output, IP/copyrights, dependency on external platforms, data/code leakage, access policies, etc.

Lack of proper governance and upskilling programs can lead to poor-quality and hallucinated code, and can make the organization vulnerable to risks like code leakage, IP issues, exposure to malicious actors, etc.

Figure 18.

More than three in five organizations have not implemented a governance framework for generative AI.

Has your organization decided on a governance framework for Gen AI for your software engineering function/process



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executives Survey, April 2024, N = 1,098 senior executives

Among the 39% of organizations with a governance framework, there remain considerable gaps:

- 61% have governance for cybersecurity issues.
- 60% have governance for productivity and quality KPIs to assess and measure generative AI output and success.
- 57% have governance for ethical issues, such as lack of transparency, bias, etc.

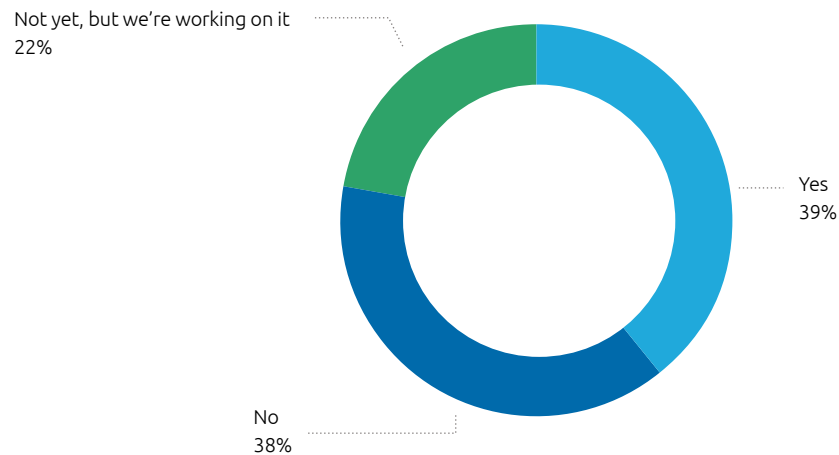
Most organizations (61%) also lack an upskilling/reskilling program (see Figure 19). Our research has shown that organizations with an upskilling program to prepare the workforce are better able to improve their workforce productivity.⁴⁰

Mousumi Bhattacharya from Centene shares, *“Generative AI in software engineering is still a new field; getting used to the tools is most important so that people are not scared of it. You need to make sure that the workforce is properly trained on the tools as well as on issues of ethics and compliance. People need to understand that these trainings will make them perform better and make them more needed in the workforce.”*

Figure 19.

Nearly three in five organizations have no upskilling/reskilling program for generative AI.

Percentage of organizations that have developed a reskilling/upskilling program for generative AI



Source: Capgemini Research Institute, Generative AI in Software Engineering, Senior Executives Survey, April 2024, N = 1,098 senior executives

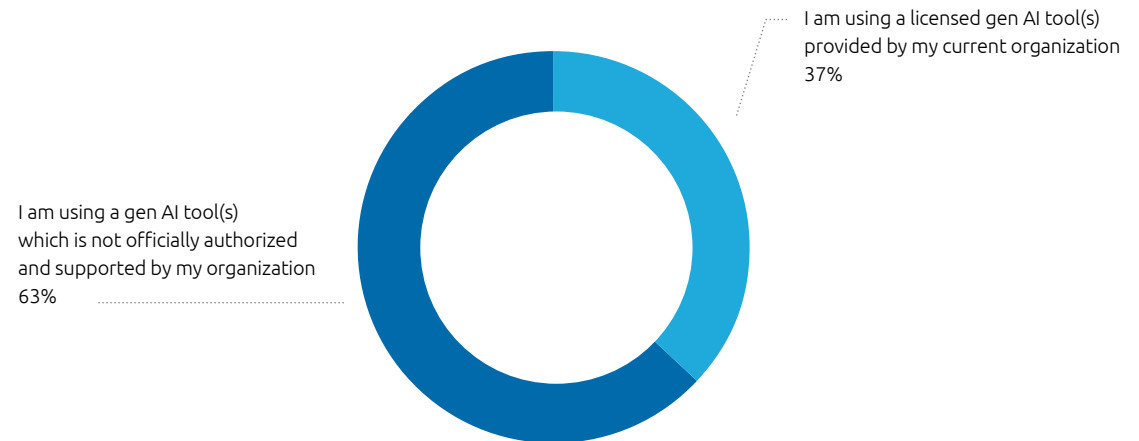
More than three in five software professionals using generative AI are doing so without organizational approval.

More than three in five software professionals (63%) using generative AI, use it in an unauthorized manner (see Figure 20). Some large organizations deploy tools and processes to check such “shadow” use. Akram Sheriff from Cisco Systems shares, *“Organizations are deploying AI-specific software firewalls to check if software professionals are using their personal access to generative AI tools to generate code. Other include allocating API keys, which give access to the Microsoft Azure OpenAI, for example, and procuring a pool of licenses to be reallocated across teams based on usage.”*

Figure 20.

More than three in five software professionals using generative AI use unauthorized tools and solutions.

Distribution of generative AI tools using workforce by the type of usage



Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professionals Survey, April 2024, N = 1,092 software professionals

Our research also shows that nearly nine in ten professionals who use generative AI in the public sector and in the insurance industry use unauthorized tools and solutions. This would lead to adverse effects without proper governance and human oversight.

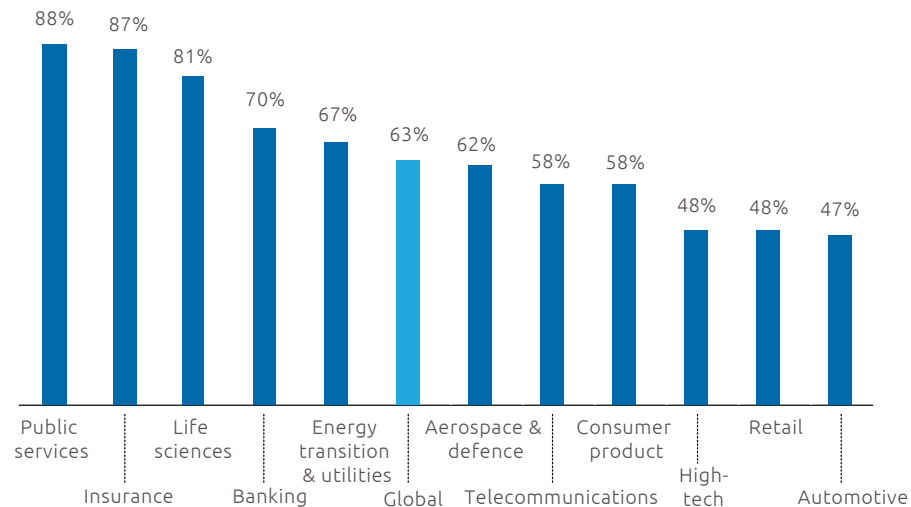
63%

Software Professionals are using generative AI tool(s) not officially authorized and supported by their organization

Figure 21.

Public service and insurance have the highest percentage of the workforce using unsupported or unauthorized generative AI tools and platforms.

Percentage of workforce using unofficial and unsupported generative AI solutions and tools, by industry



Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professionals Survey, April 2024, N = 1,092 software professionals

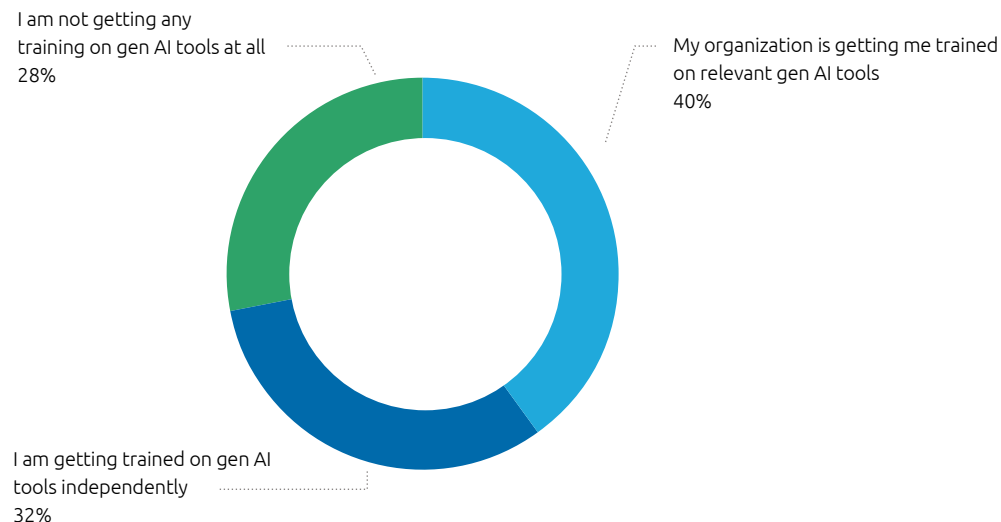
The workforce needs to rely on self-training for generative AI, as only two in five organizations are upskilling their employees.

Software professionals are upskilling themselves independently on generative AI tools, covering the lack of organizational support. As shown in Figure 22, not more than 40% of software professionals are getting trained by their organizations on relevant generative AI tools.

Figure 22.

A third of software professionals are training themselves independently on generative AI tools

Distribution of software professionals based on the training they receive



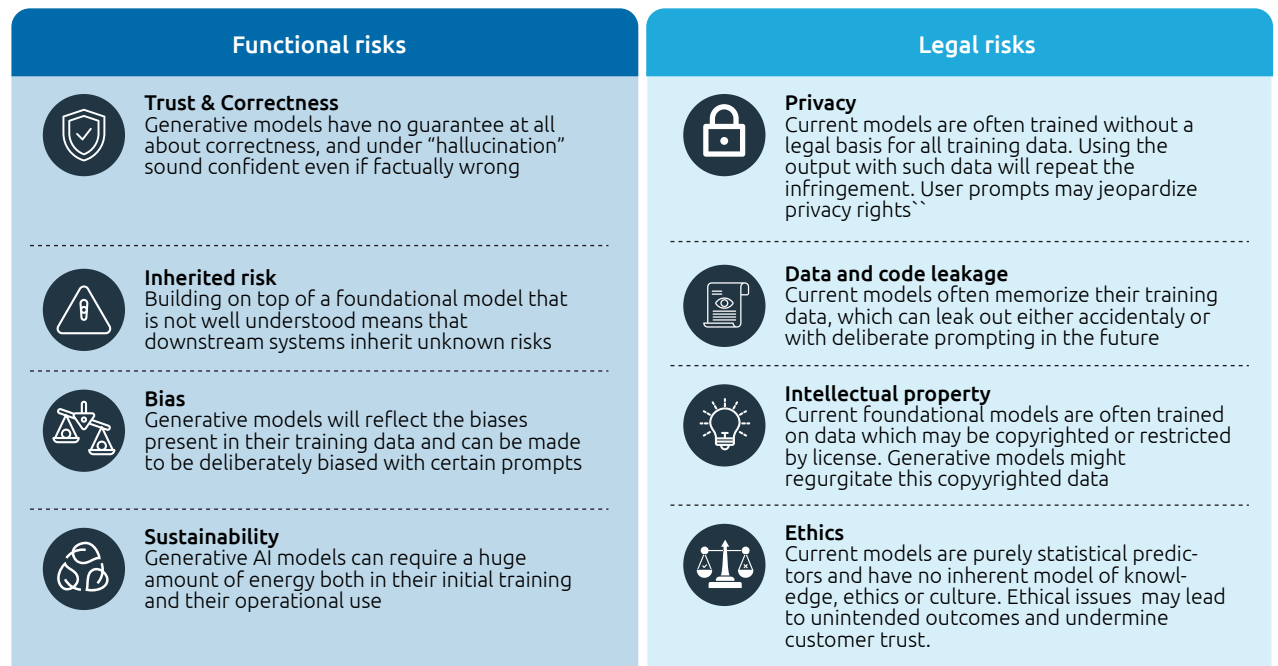
Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professionals Survey,, April 2024, N = 1,092 software professionals.

Using unapproved tools without proper training and safeguards exposes organizations to risk

As seen in Figure 23, legal concerns, functional risks, and data security concerns are the top challenges senior executives associate with implementing generative AI. A recent breach, where employees accidentally shared a piece of source code in the public domain, illustrates these risks.

Figure 23.

Various functional and legal risks that underlie generative AI use in an organization.



Source: Capgemini Research Institute analysis.

With software professionals adopting generative AI tools without organizational approval and oversight, these challenges increase. Two-thirds of our respondents agree that generative AI can bring unintended consequences unless properly validated through a regression test framework. Even more agree that unsupervised generative AI output can lead to complicated debugging, edge cases, and bugs, which can disrupt project timelines and delivery.

The following are key risks of adopting generative AI without proper governance, guidelines, and training:

32%

Software professionals are getting trained on generative AI tools independently

Security vulnerabilities: Code generated with unauthorized AI tools expose internal systems and applications to malicious actors, which can lead to data leakage, unauthorized access, and cyberattacks. It may also expose the internal code to competitors and third parties by making it part of the training data.

Reliability concerns: Ensuring reliability and quality of the generated output is a challenge when employees use unauthorized tools without proper validation and system checks. Organizations should double down on data governance and prioritize visibility regarding the quality and clarity of the provenance of any data used.

Potential legal issues: If AI models are trained on copyright or proprietary code or data, this may cause legal complications and IP issues. A class-action suit has been filed against Microsoft and GitHub, stating that the tool predictively generates code based on what the programmer has already written. The plaintiffs allege that Copilot copies and republishes code from GitHub, disregarding the requirements of the latter's open-source license.⁴¹

When an organization officially approves the use of generative AI, it would generally address such aspects. But individual employees may well overlook these issues without proper training and education on the reliability, security, legal, and ethical aspects of using generative AI.





04

How can organizations harness the full potential of generative AI for software engineering?

Considering the outcomes of the research and our experience of helping clients experiment with, scale, and achieve more with generative AI for software engineering, we believe that a calculated and step-by-step approach, as outlined below, has a high chance of harnessing generative AI's full potential.

■ Select and prioritize high benefit use cases.

Organizations must prioritize these for quick wins. Use case with high potential for productivity improvement and other benefits such as higher quality among others, include:

- Creating literature and documentation
- Coding assistance
- Debugging and testing
- Identifying security vulnerabilities
- Code modernization (including code translation, code migration, and code conversion)
- RFP preparation and evaluation

The first task is to assess the current software engineering lifecycle to highlight key opportunities to generate value for the software organization and workforce. Real experimentation is the bedrock of this implementation exercise, starting with a few teams, then extending across multiple domains in the organization. In parallel, a governance process involving key stakeholders from across IT and business teams must decide which use cases to prioritize, based on their risk-weighted rewards (more on this in the next point). The governing body must include a central design authority for designing solutions, controlling their implementation and progressive deployment.

39%

of organizations are focusing on code assistance despite it being high benefit use cases

“Generative AI for software engineering is still a new field; getting used to the tools is most important so that people are not scared of it. You need to make sure that the workforce is properly trained on the tools as well as on issues of ethics and compliance. People need to understand that these trainings will make them perform better and make them more needed in the workforce.”

Mousumi Bhattacharya

Director of IT at Centene, a US-based managed care company

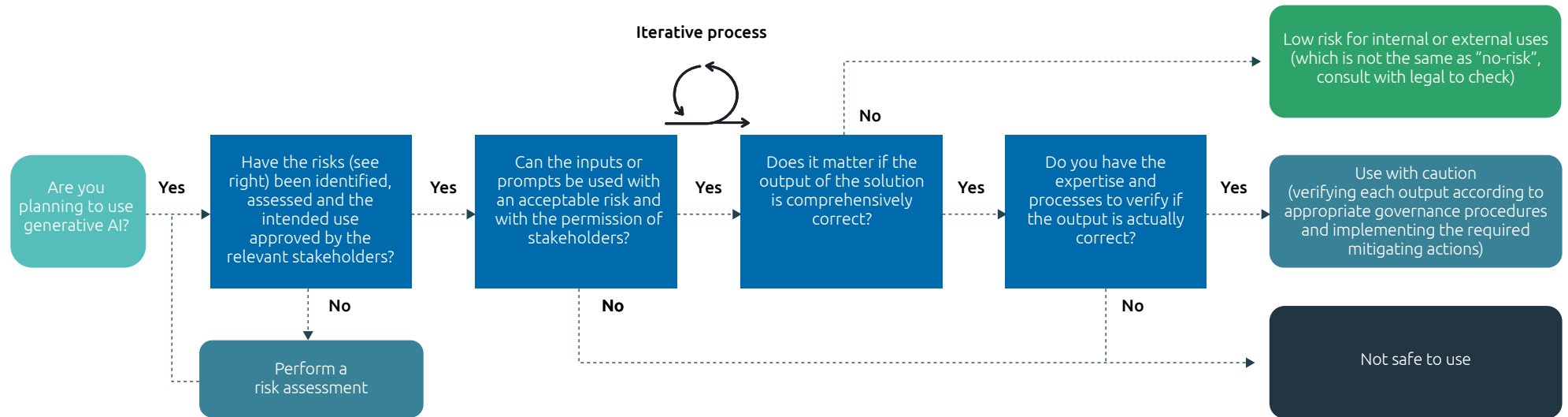
- **Mitigate risks around security, IP/copyright issues, and code leakage using a thorough risk management approach.**

Generative AI use entails risk, which must be addressed at the planning stage. For each use case selected, you should start with a risk assessment (see Figure 24) and involve your organization’s legal, IP, cybersecurity, and data protection experts early on.

Involve and inform other stakeholders (across business, strategy, product, marketing, and sales, as relevant) about potential risks, ascertain an acceptable risk level, and conduct a scenario analysis to simulate outcomes.

Figure 24.

A risk assessment framework to kickstart generative AI implementation in software engineering



A software architect from a leading multinational bank says, *“As we move from proof-of-concept (POC) to the roll out phase, assessing and documenting the risks becomes important. Selecting the right code for generative AI to work on, along with the right checks in place can help mitigate such risks like proprietary code leakage.”*

From a cybersecurity perspective, most of the existing application security policies still apply to generative AI applications. Some of the important and relevant application security practices to follow are:

- Use threat modelling to identify the security gaps as generative AI is more likely to use modern development approaches (such as DevOps) and architectures (such as service mesh).
- Preventing the usage of libraries, frameworks and languages with known vulnerabilities from unknown sources by enforcing strict security standards and regular audits.
- Mandating secure coding standards by frequently scanning the code for known vulnerabilities
- Ensuring data security, especially the use of sensitive data during the development process. For example, during coding use cases it is integral to prevent leakage of secrets in public code repositories. Use automated quality checks for secret detection scanning and manual review.
- To guard against the risks posed by hallucination of generative AI, software professionals must exercise extra caution to recognize these inaccuracies. They must always review code and perform code scanning through built-in tools or third-party tools that detect vulnerabilities or incorrect code. Using prompt engineering, Retrieval-augmented generation (RAG), or fine-tuning can also reduce risks of hallucination besides making the code more contextual and relevant. More on this in the points that follow.

While these practices are necessary, they may not be sufficient to guard against all risks. The first line of defense starts with people. Users should be trained enough to use a generative AI system responsibly and have enough expertise to review the output for risks.

■ Transform your software organization to ensure optimal usage of generative AI.

Traditional approaches focus on incremental improvements in lifecycle productivity and quality. Generative AI allows us to completely transform the traditional approach by asking questions:

- How to understand the full impact of generative AI on the software organization?
- How to select and plan the right use cases that have highest benefit yet ease of implementation?

- How to measure (and thus, manage) value realization from the start?
- How to contextualize generative AI tools with organization's internal or proprietary content?

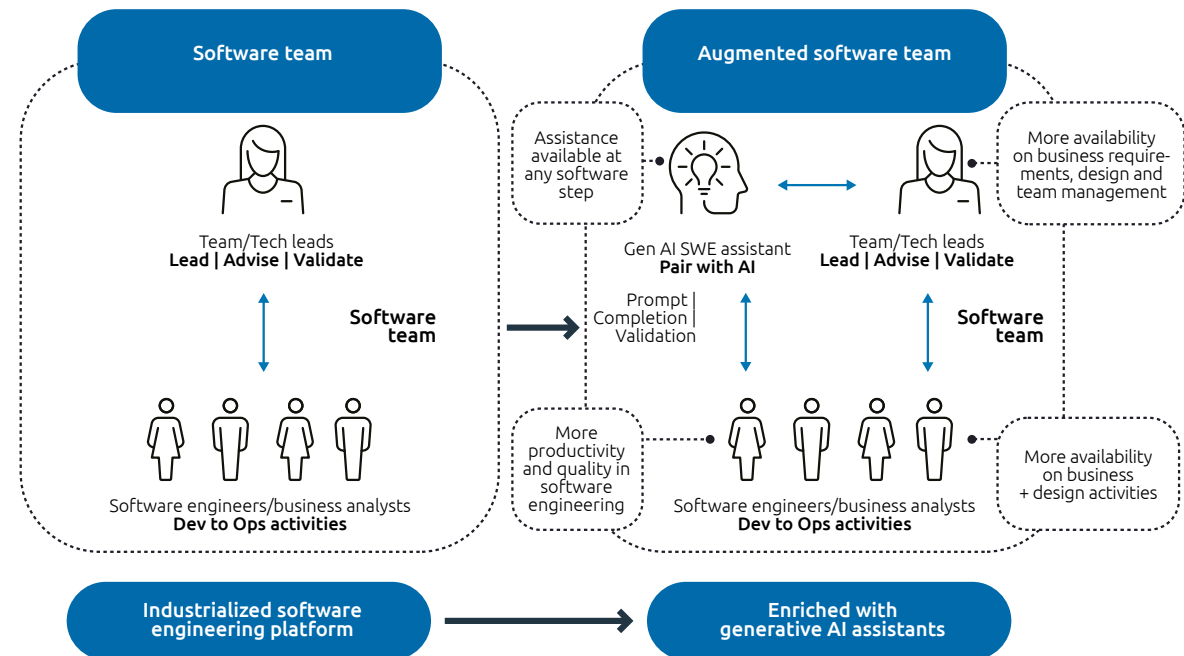
Attempting to answer these questions holds keys to harnessing the most potential of generative AI, multiplying software teams' productivity while also delivering on quality, security, and other important parameters. Here are a few ways to make it happen:

• Augment your software teams with a new member – a generative AI assistant.

Generative AI has applications and benefits across the SDLC, and the accuracy and usefulness of its output is continually increasing. Why not embed generative AI assistants in every software team? AI-augmented software teams can automate mundane tasks, expedite debugging, and promote more nuanced collaboration, thereby optimizing the software lifecycle productivity and improving learning curves both the human and AI side (see Figure 25).

Figure 25.

Concept of a generative AI-augmented software team.



Source: Capgemini Research Institute analysis.

Every professional on an augmented software team can benefit from the generative AI assistant – junior or senior. In our survey, a majority of junior (53%) as well as senior professionals (58%) believe that generative AI tools will augment their day-to-day work within the next two years. For instance, for junior developers, generative AI tools offer a faster route to learning new programming languages and techniques that they may not be familiar with and pick up nuances of programming in the organization's context. Whereas senior developers can focus on grooming juniors, sharing their experience and ensuring that juniors are learning and retaining their knowledge, reviewing output of junior developers developed independently or with generative AI assistance, solving complex issues, and collaborating with business counterparts to drive higher value.

This approach can help software teams retain the foundation principles of Agile and DevOps – collaboration, adaptability, time to value, product-centricity, and continuous feedback loops – while benefiting from new ways of working, such as:

- AI–human collaboration:** Team members work individually with the AI assistant to automate repetitive tasks, understand and solve problems, or brainstorm ideas, leveraging an LLM's speed and knowledge while ensuring human creativity. It can also assist in training junior professionals. As shared by Director of Product Marketing at a multinational technology major, *"This is a great onboarding and training tool to help the new hires get up to speed on how to use the company's software language. This is one of the areas where we found it to be immensely beneficial."*
- Cross-functional collaboration:** Generative AI assistants help cross-functional teams (such as IT and business) collaborate seamlessly by helping them communicate in each other's language.
- Senior coordination:** A senior lead oversees the augmented team, coordinating efforts and resolving conflicts while ensuring control and validation to promote a smooth workflow and healthy team dynamic.
- Augmented pair programming:** The basic unit of an augmented software team is pairs of software engineers working intermittently with each other or individually with the AI assistant, each helping the other. Software professionals review the generated output and enhance its quality.



- **Identify requirements for new capabilities and source them.**

As generative AI performs routine tasks well, it necessitates a rethink of established processes. New capabilities will be needed. As a senior director from a global pharmaceutical major put it, *“Software teams would need to develop advanced understanding around how generative AI works, understand its strengths and weaknesses and distinguish smartly where generative AI will work and where it won’t. Those who crack this code can turn into super developers with generative AI!”*

For example:

- **Ability to pair with AI:** The most immediate new capability needed is the ability to interact with generative AI assistants to analyze their output, validate it or iterate on it until an acceptable solution is obtained. More experienced software professionals will find it helpful working hand in hand with a generative AI assistant, much like “pair programming” in which two programmers work together on one workstation – one programmer writing code, while the other reviewing the code as it is typed in.
- **Leading generative AI assistants/agents and overseeing their work:** As generative AI assistants evolve into autonomous agents capable of reliably assisting on specific, routine tasks end-to-end, human teams must oversee multiple AI agents, coordinating their output and

resolving/preventing conflicts. For team leads and project managers, it translates to a greater focus on business requirements, design, and team management.

- **Developing new resources** such as prompt libraries and playbooks for software teams and keeping them up to date. A prompt library is a collection of detailed and documented prompt engineering patterns to apply across the full software development lifecycle, further saving software professionals’ time to come up with useful and efficient prompts for generative AI assistants. Playbooks are guideline documents for software teams’ reference to support and advise on secure ways of working with generative AI assistants. Organizations will also have to contextualize the generative AI assistants and extend their functionality using their private, sanitized data for use cases under trial/implementation. This would lead to more relevant, robust, and efficient assistance from generative AI.
- **Fixing complex bugs and root-cause analysis of problems:** Human software teams will still be needed to fix complex coding problems given their inherent advantage and analyze root causes of issues that generative AI can’t solve or prevent. This capability will only grow in future as human programmers’ time is redirected towards complex problems, customer interaction, and debugging.

- **Prepare for generative AI use by delivering technology prerequisites.**

- **Build a repository of platforms and tools for a seamless and augmented software engineering experience.**

The biggest gap in essential prerequisites is usually access to platforms and tools, including integrated development environments (IDEs), automation and testing tools, and collaboration tools (only 27% of organizations claim to have above-average availability of these). Of testing domain

27%

Organizations have the platforms and tools to implement and scale generative AI for software engineering

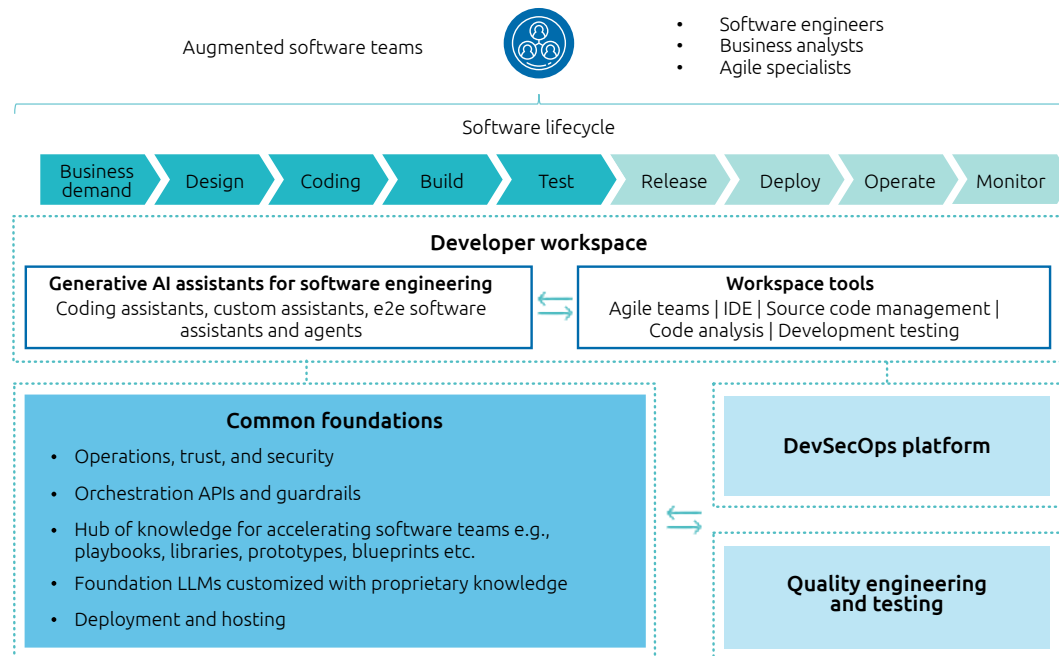
professionals, 24% say they have access to these tools, compared to 19% of project and program management professionals, further highlighting gaps within the software engineering function. In addition, only about half of the organizations surveyed have access to the computing infrastructure and support needed to fully leverage generative AI tools. Here again, significantly fewer (40%) project and program management professionals claim to have this access.

Figure 26 shows three key platforms and associated tools that can help provide a seamless and augmented experience to software teams through all phases of SDLC and across the organization:

- **Developer workspace:** Provides an interface to the generative AI assistance as part of existing or new workspace tools for IDEs, source code management, etc.
- **Common foundations:** Host foundational LLMs customized with proprietary data, APIs, and playbooks to accelerate generative AI deployment.
- **DevSecOps platform:** Helps software developers and operations teams perform security testing and evaluation at every while also interfacing with quality engineering and testing.

Figure 26.

A reference framework of foundational platforms and tools for a seamless software engineering experience



Source: Capgemini Research Institute analysis.

For the software workforce to make optimal use of generative AI, they need these tools seamlessly integrated into their workflow. The Chief Business Digital Officer at a leading automaker said: *"It's important that these tools integrate seamlessly because IT systems are already very complex. We look for tools that can integrate with our development environments."*

Amazon Q Developer, AWS's generative AI assistant for software development, which was recently made "generally available," offers inline code suggestions and chat in popular IDEs, including JetBrains, IntelliJ IDEA, Visual Studio, and VS Code, and across 15 coding languages.⁴² This smoothens a developer's workflow, making code generation and debugging faster. BT Group used Amazon's generative AI coding companion, generating over 100,000 lines of code in the first four months. The solution provided 15–20 code suggestions per user per day, of which 37% were accepted by software engineers.

- **Privately and safely contextualizing generative AI assistants with an organization's own content**

Large language models that form the foundations of today's generative AI assistants are generic and do not internalize enough information about the specific context in which they are used. This creates code issues and inefficiencies leading to less relevant code and related content creation and, as a result, limits improvements in productivity and quality. This limitation is more evident in use cases requiring access to enterprise data (enterprise content documentation repository, legal data/document repository, business documentation and design requirements, among others) to contextualize the model for specific desired outcomes. Archisman Munshi, co-founder of a senior citizen care company that is tackling financial fraud issues using AI elaborates, *"Each company has its own way coding style, standards, patterns, and associated guardrails. Properly training the LLM with our data and processes ensure more robust code suggestions which are better aligned to an organization's conventions and guidelines, thus minimizing inefficiencies in the code."*

Retrieval-augmented generation or fine-tuning are needed to fully adapt the generic models to the specific context or problem domain where they are being applied. Contextualizing generative AI output like this improves the quality of its output significantly by improving the accuracy and relevance of responses. Consequently, users' productivity is boosted as well since less time is spent looking for organization's internal specifics such as terminologies, conventions, processes etc.

There are two major approaches for privately and safely customizing generative AI assistants for organizations' specific context using proprietary or enterprise data and content:

1. Retrieval-augmented generation (RAG): This approach combines a question/answer system (like a search engine) that fetches relevant document chunks from a large corpus (say internal knowledge base), and an LLM, which produces answers for the user using the information from those chunks.
2. Fine-tuning: This is the process of taking a foundational LLM and further training it on a smaller, specific dataset to adapt it to a particular task/domain and improve its performance.

■ Adopt a measurement protocol for generative AI impact monitoring and use case prioritization.

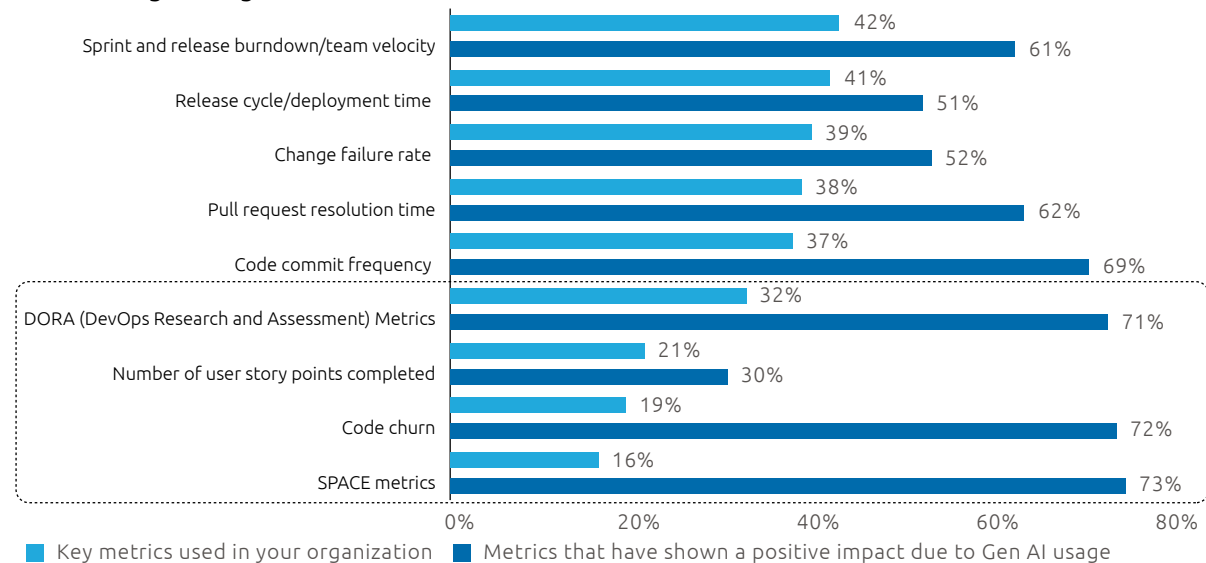
What gets measured, gets managed – might be an old saying but rings truer in new paradigms that generative AI is unfolding. Almost half of organizations in our survey (48%) have no standard metrics to gauge the success of generative AI use in software engineering (see Figure 27). Our survey also reveals that the commonly used metrics, while suitable for regular productivity measures, such as time to deploy or to resolve issues, do not fully capture the benefits of generative AI, especially on non-conventional measures of productivity, such as employee satisfaction. These are better captured by metrics frameworks like DORA and SPACE.⁴³ Although these metrics are yet to gain traction, as they are costly and time-consuming to implement. This finding indicates that a set of metrics including KPIs for velocity, quality, security, and developer experience can prove useful.

Measuring productivity within software engineering poses inherent complexities due to the multifaceted nature of the SDLC, the dynamic environment it operates in, and the subjective and intangible aspects of many

Figure 27.

Most organizations show improvement from use of generative AI when measured using less popular, but more holistic, productivity metrics frameworks, such as SPACE and DORA

Metrics predominantly used vs. Metrics impacted through use of Generative AI for software engineering



Source: Capgemini Research Institute, Generative AI in Software Engineering, Software Professionals Survey, April 2024, N = 790 respondents overall. The question on 'Metrics that have shown an impact with Generative AI' has been answered by the subset of respondents who chose the respective metric as a key metric used in their organization.

of its components. A comprehensive measurement approach must encompass both qualitative and quantitative factors, tailored to the specific context of the project. Similarly, assessing software quality presents challenges as it involves various dimensions such as functionality, performance, reliability, usability, maintainability, security, and scalability, each requiring its own set of metrics and criteria.

We recommend that organizations adopt a measurement protocol involving a measurement approach, metrics, team, and a well-defined process, to create actionable and reliable results:

1. **Measurement approach:** It includes steps for measuring progress, such as: planning, setting baselines, and running an experimentation. Once the right metrics have been identified (more on this in the next point), tools like SonarQube, CAST, Jira, and developer surveys are used for collecting and analyzing data. Team stability, duration, technology, legal considerations, and cybersecurity are key prerequisites and success factors for a consistent measurement approach.
2. **Metrics:** Coding velocity serves as a pivotal metric to measure team productivity, focusing on coding and unit testing activities, typically measured by

implemented story points. Moreover, coding velocity per developer capability or seniority level discerns productivity variances, comparing time taken to complete a set number of story points with and without generative AI assistance. This approach offers insights into how generative AI influences productivity across different developer skill levels and the intricacies of software development tasks. Test coverage, code efficiency, code security, code smells, code duplication among others, serve as essential KPIs for relevant use cases.

3. **Team:** In the single team measurement approach, one team sequentially executes a backlog of user stories of consistent size or complexity, comparing performance with and without generative AI assistance. On the other hand, a multiple team's approach involves parallel execution of the same backlog by at least two teams with different tool setups (e.g., with and without generative AI tools), allowing for simultaneous assessment of generative AI's effects across different team dynamics. The seniority or capabilities of a team are important for normalization, thus it's mandatory to know what kind of team mix is working on the defined backlog.

4. **Process:** Once all the ingredients have been defined, a process is needed to ensure high quality results and reduce side effects due to estimation inaccuracy and human factors:

- Define the team(s) organization and the experimentation scope and timeline.
- Define the measurement approach.
- Validate the prerequisites and success factors.
- Conduct a baseline for the metrics, without generative AI assistance.
- Execute the experimentation sprints with generative AI assistance.
- Collect and normalize the metrics and the feedback.
- Consolidate and report the measured results.

■ Put people at the heart of this transformation by creating a learning culture at your organization.

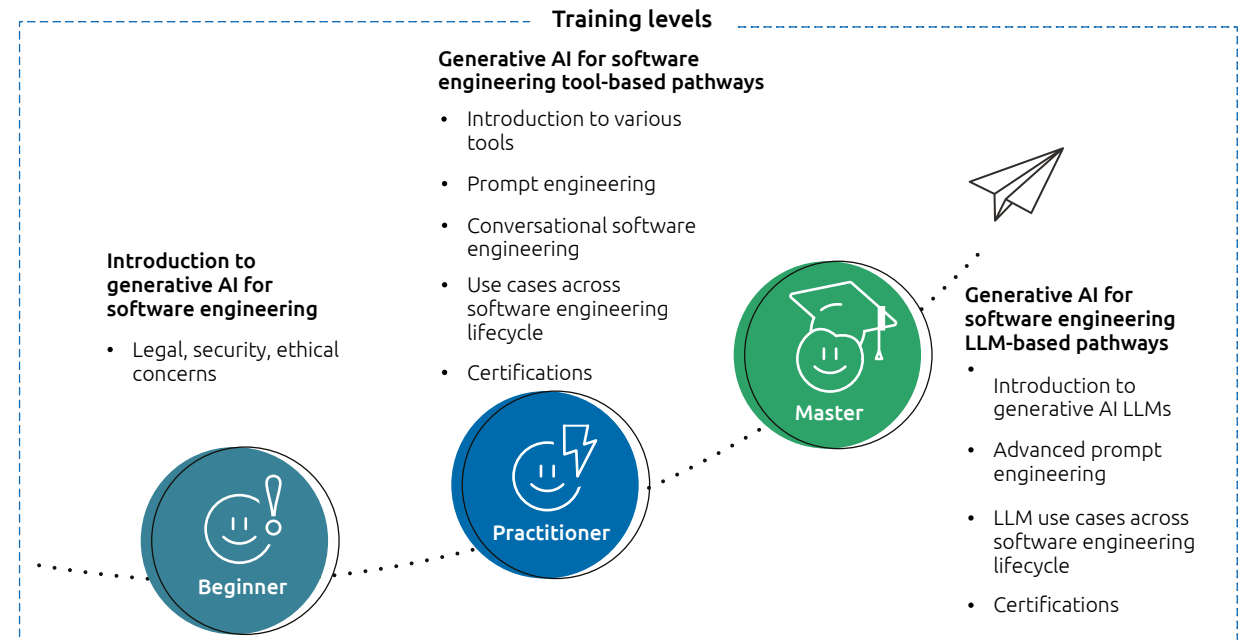
- Provide upskilling and cross-skilling opportunities.

As we saw earlier, a sizeable share of the workforce (41%) is rushing ahead to equip themselves with skills to use generative AI tools for software engineering. 40% of these “independent learners” are doing so through paid courses highlighting the importance of the training for them. These are key indicators of how organizations are lagging in providing upskilling opportunities to their workforce.

Figure 28 shows a blueprint for a training program to build generative AI skills in the software workforce. Indeed, 51% of senior executives believe that leveraging

Figure 28.

Blueprint of a training program to impart generative AI skills to software teams.



Source: Capgemini Research Institute analysis.

generative AI in software engineering will require significant investment in upskilling and cross-skilling the software workforce. Yet only 39% of organizations have a generative AI upskilling program for software engineering. Moreover, current upskilling programs are ill-equipped to provide training in areas of evaluation of generative AI code/output (56%), security (43%), and IP and legal issues (33%).

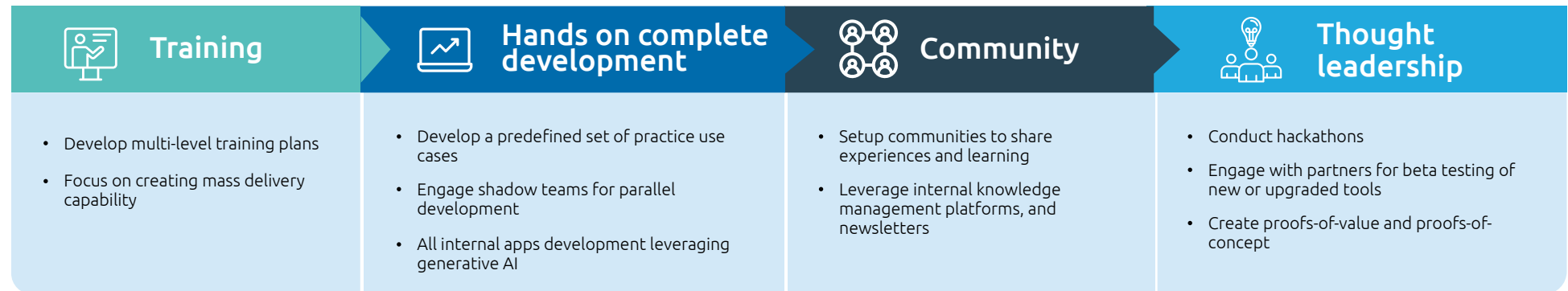
Our survey data reveals that workforces at organizations that provide training and access to generative AI are more optimistic about their productivity (84% vs. 74%), job satisfaction (43% vs. 32%), and future employability and pay (35% vs. 29%) than those at organizations without such training and access. They also report being happier (34% vs.

23%) and less negative about the adoption of generative AI for software engineering (6% vs. 22%).

Having a training and upskilling program is just the first step. They can prove to be insufficient and rendered irrelevant soon if not backed by systems and processes that create a culture of learning and thought leadership within the organization adopting generative AI for software. Figure 29 shows a blueprint of a mobilization plan to develop generative AI delivery capability.

Figure 29.

A mobilization plan to develop generative AI delivery capability.



Source: Capgemini Research Institute analysis.

- **Address employees' work displacement concerns**

44% of senior executives cite risks with displacing and transitioning of workers as the biggest challenge for generative AI implementation. It is important to make the workforce understand the role of AI as a tool that 'augments' and 'assists' and does not 'replace.' Archisman Munshi agrees, *"It is imperative to educate people to view AI as an 'assistant' that will make it easier and faster to complete their day-to-day tasks. Also, people won't get replaced, it is the tasks that will get replaced. Instead of spending time on routine and mundane tasks like documentation, people will now get to spend more time on value-adding and challenging work."*



"It is imperative to educate people to view AI as an 'assistant' that will make it easier and faster to complete their day-to-day tasks."

Archisman Munshi,

Co-founder of a senior citizen care company that is tackling financial fraud issues using AI

Conclusion

Generative AI offers a new way to augment software engineering by boosting new software development, software quality, and software teams' productivity. While it is still early days in the trial and actual use of this technology, benefits are already being seen. However, as of today, more software professionals are using generative AI tools using unauthorized ways than via authorized access. If left unchecked, this unauthorized usage can expose organizations

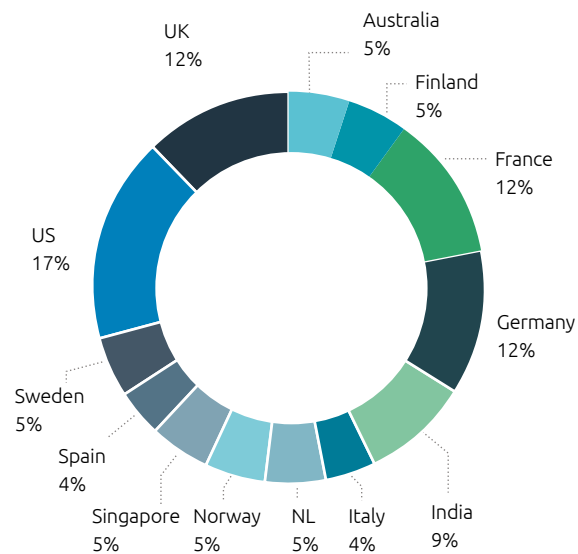
to various kinds of risks and damages. Leading organizations are able to manage this risk and yet derive the maximum potential generative AI has to offer today by transforming their software organization, delivering technology prerequisites, measuring the realized value, and taking their people along on this transformative journey.

Research methodology

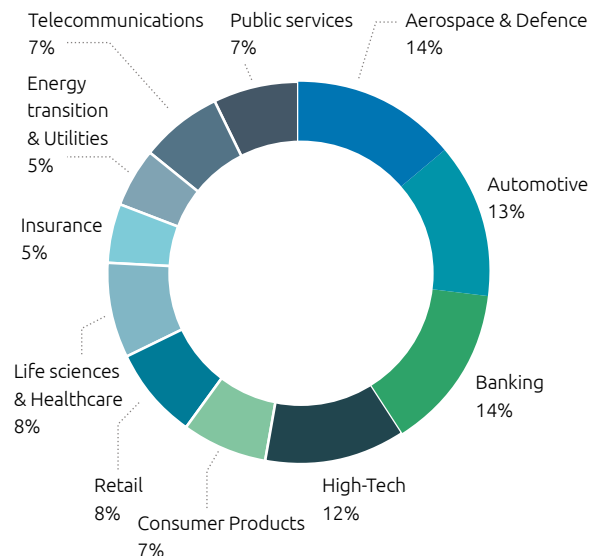
We surveyed 1,098 senior executives (director and above) and 1,092 software professionals (architects, developers, testers, and project managers, among others). We also conducted 20 in-depth interviews with leaders from clients, partners, and startups, along with several professionals.

Nearly all of the software professionals and senior executives represent a similar set of organization.

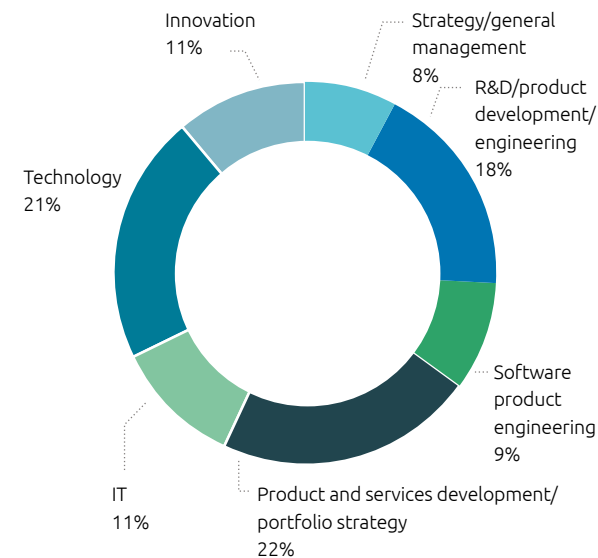
Senior executives, by country of organization's headquarters



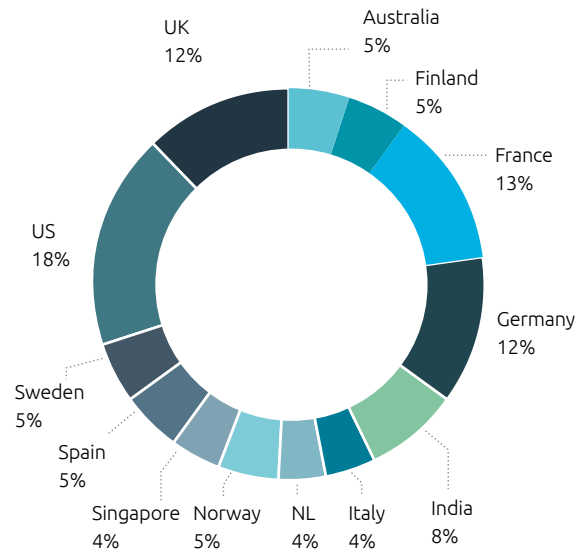
Senior executives, by sector



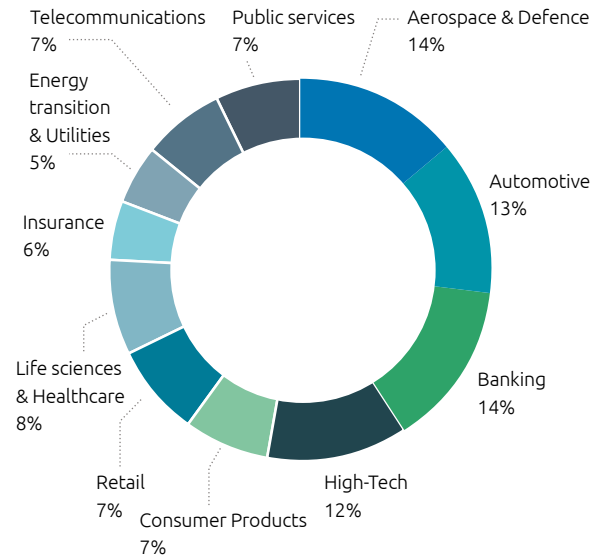
Senior executives, by function



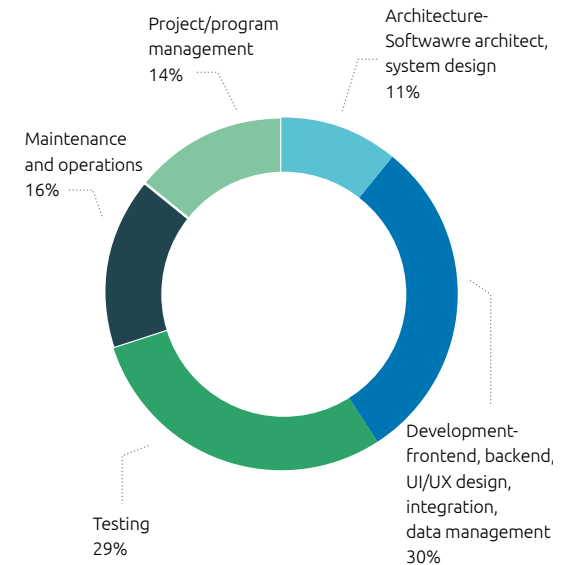
Software professionals, by country of organization's headquarters



Software professionals, by sector



Software professionals, by software domain



Appendix

List of use cases analyzed in this research:

| Phases of SDLC | Use cases |
|------------------------|---|
| Business Demand | Business requirement/demand analysis and writing |
| | RFP preparation and proposal evaluation |
| | User story generation |
| Design | Architecture writing assistance and modelling |
| | UX/UI design assistance |
| Coding | Coding assistance (including code generation, completion, etc.) |
| | Debugging and error prediction |
| | Unit test generation |
| | Code explanation |
| | Legacy code modernization (migration, conversion, etc.) |
| | Identification of software security vulnerabilities |

Continue to the next page...

| Phases of SDLC | Use cases |
|----------------|---|
| Build | Application integration |
| Test | Test case generation/ testing code generation |
| | User acceptance testing |
| | Code review and quality assurance |
| Deploy | Platform provisioning and configuration (Infrastructure as code script writing support) |
| Monitor | Incident analysis and resolution |
| | Log analysis |
| | Project management and collaboration (e.g., backlog planning, effort estimations, team effectiveness analysis, communication) |
| | Agile ceremonies |
| | Documentation |

As a worldwide leader, generative AI for Software Engineering is strategic and core for our business to deliver the best value to our clients

Capgemini is a global leader with over 100K developers specializing in custom software engineering. We identify and evaluate the potential best generative AI use cases for software engineering in order to create custom, innovative solutions. We rigorously assess and test through rigorous pilot programs with industrialized and standardized protocols to measure impact. Once proven, we seamlessly integrate them into our delivery processes to ensure we maximize value for our clients.

- **Evaluate Market Tools:** We select the most suitable tools for each phase of the Software Development Life Cycle (SDLC).
- **Legal and Compliance Framework:** We develop frameworks to understand and meet clients' specific legal and compliance requirements, ensuring secure and safe generative AI usage.
- **Impact Assessment:** We analyse the impact on team organization, structure, skills, and workflows.
- **Solution, Asset and Method Development:** We build solutions, assets and methods to address software engineering needs, ways of working and enhance existing market tools throughout the SDLC.
- **Measurement Protocols:** We implement an auditable measurement protocol to evaluate the impact of generative AI on software engineering across the SDLC.
- **Global Partnerships:** We leverage our global partnerships with leading technology and tool vendors.
- **Value Generation:** We utilize generative AI in our projects to deliver enhanced value to our clients

Leader in **designing & managing complex IT transformations**, engaging CIO / CxOs



Global Leader in **Software Engineering**, with **100,000+** experts

Capgemini's strategic partnerships on generative AI



Expected outcomes

Productivity Gain

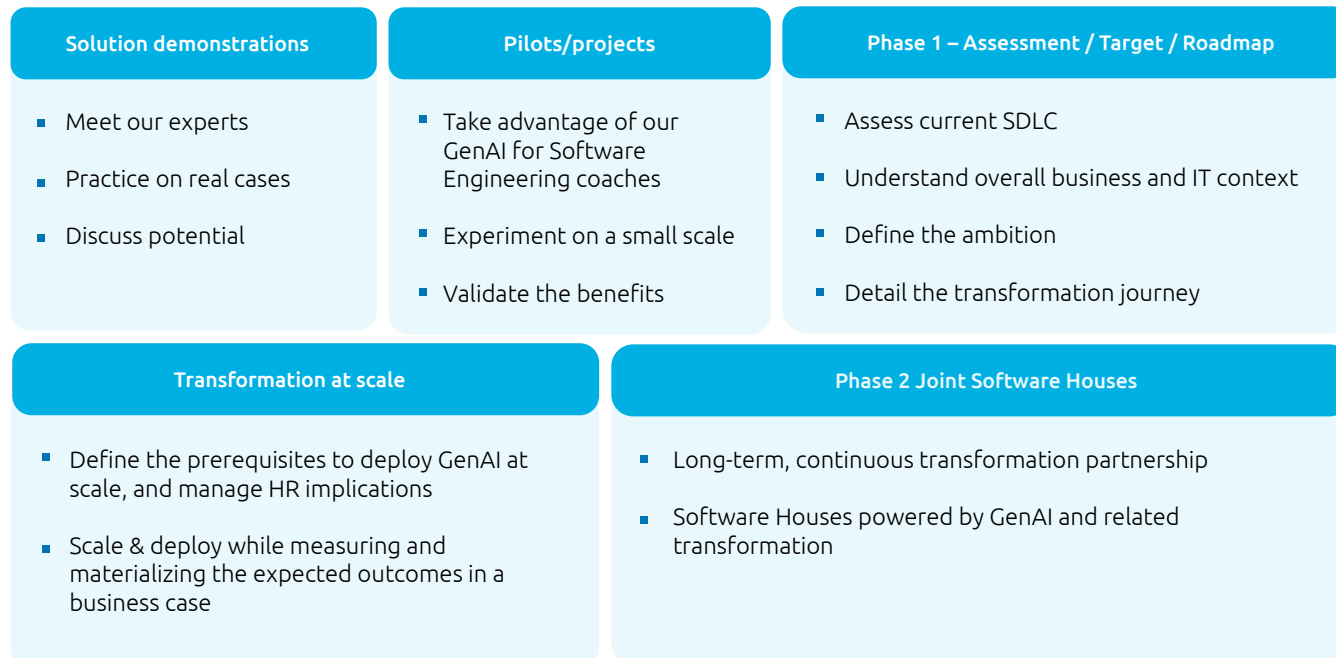
Quality and Security

Time to Market

Developer Experience

Maximizing Software Engineering Transformation with generative AI

In addition to utilizing generative AI for software engineering in client projects, we have a comprehensive strategy to guarantee our clients maximize the benefits of their generative AI driven software engineering transformation.



Accelerate Software Quality Engineering with generative AI

With the power of Gen AI, Sogeti, part of Capgemini, has embedded our quality engineering expertise and proven methodology, TMap®, into a ready-to-deploy platform.

We view Quality Engineering as an end-to-end practice—requiring consistent, standards-compliant artifact creation throughout the software creation process.

By using Gen AI Amplifier for our clients, we are accelerating and improving the effectiveness of quality engineering for their applications from the start – and at every step of the way.

As Quality Engineering & Testing experts, we have built Gen AI Amplifier with rigorous safeguards, infused our proven

best practices, and implemented a user interface that aligns with standard software quality engineering processes to ensure consistent, reliable results that meet or exceed industry standards.

What can it do for you:

Accelerate drafting of traceable requirements from conversation transcripts, creation of INVEST-compliant user stories, definition of efficient test cases, generation of code for automated test scripts, and synthesizing of relevant & context-aware test data.

How can you get started



Option 1

Book a Demo

A 1–2 hour demonstration of our Gen AI Amplifier use-cases with standard input examples.



Option 2

Evaluate proof of value

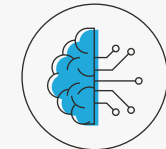
Over 2-weeks we'll help you understand the potential project impact based on your real input data.



Option 3

Run a pilot

Over 8-weeks we'll perform in-depth productivity analysis across teams and potential new use cases.



Option 4

Just get started

We'll get Gen AI Amplifier fully integrated and operational, ready to enhance your speed and productivity.

How generative AI can help you explore this report

Step 1

Share the report with your generative AI model (upload or link) and ask for support in exploring it considering your role, industry and type of organization you belong to.

For example, "I'm the CIO of a large US-based automotive company and I'd like your support in exploring this report [upload or link] to extract relevant findings."

Step 2

Prompt your generative AI model drawing inspiration from the following examples of simple prompts that can be copied and pasted into your model's chat.

- "List the key findings and key recommendations of the report"
- "Provide the list of most adopted use cases with short descriptions as referenced in the report"
- "Explain the key benefits of leveraging generative AI for software engineering according to the report"
- "Explain the best approaches for organizations to prioritize and select high benefit use cases in generative AI implementation for software engineering"
- "List the key metrics and KPIs that can be used to measure the impact of generative AI implementation in software engineering"
- "Describe the potential risks associated with using unauthorized generative AI tools for software engineering, and how to mitigate these risks"

References

1. By total productivity improvement we mean overall improvement in the productivity of the individual from all types of tasks accelerated by generative AI.
2. Includes code generation, completion, etc.
3. <https://x.com/karpathy/status/1617979122625712128>
4. American Banker, "AI will make workers 'superhuman': Goldman Sachs CIO Marco Argenti," March 16, 2023.
5. Capgemini Research Institute, "The art of software: the new route to value creation across industries," November 2023
6. GitHub, "Emirates NBD delivers cutting-edge digital banking with GitHub Copilot," 2024.
7. Technology magazine, "How BT Group is Deploying Gen AI Coding Solutions With AWS," Feb 21, 2024
8. WSJ, "Can AI Solve Legacy Tech Problems? Companies Are Putting It to the Test," January 26, 2024.
9. AWS, "Amazon Q Developer customers," 2024.
10. CIO.com, "The early returns on gen AI for software development," March 12, 2024.
11. Cio.com, "The early returns on gen AI for software development," March 2024
12. Github.com, Github Copilot, 2024.
13. World Economic Forum, "Generative AI for small-medium-sized business: cybersecurity chaos or empowerment?" June 20, 2023.
14. codeium.com, Accessed on June 12, 2024.
15. Microsoft, "Redesigning how we work at Microsoft with generative AI, March 5, 2024.
16. Microsoft, "Siemens connects frontline workers and engineers for real-time problem-solving using Azure AI," November 2023.
17. Atlassian Marketplace, "AI Test Case and User Story Issue Auto Generator for Jira," accessed June 2024.
18. IBM, "Modernization and code generation," Accessed on June 12, 2024.
19. Financial Express, "Simple Terms: Generative AI For Software Engineering," March 14, 2024.
20. WSJ, "Can AI Solve Legacy Tech Problems? Companies Are Putting It to the Test," January 26, 2024.
21. Github.com
22. Google, "Generative AI's impact on developer productivity."
23. Amazon Q Developer, "Write code and build applications faster and easier".

24. Github, Sketch2Code
25. PRNewswire, "Archie Named Top AI & Data Product in the 2024 Product Awards," 2024.
26. WritemyPRD.com
27. Eraser.io
28. Mintify.com
29. Diffblue.com
30. Synk.io
31. Sapien.ai
32. Bloop.ai
33. OpenText, "Use AI to cut software delivery time by up to 60%."
34. GitHub, "Duolingo empowers its engineers to be force multipliers for expertise with GitHub Copilot and Code spaces," 2024.
35. Diffblue, "Goldman Sachs complete a year's worth of Java unit test writing overnight with Diffblue Cover," 2024
36. Sourcegraph, "Qualtrics speeds up unit test creation and understanding code with Cody", 2024
37. WSJ, "Can AI Solve Legacy Tech Problems? Companies Are Putting It to the Test," January 26, 2024.
38. The views expressed are the independent views of the respondent and do not reflect the views of any organization.
39. Sourcegraph, "Cody + Leidos: Maximizing efficiency with heightened security in the AI race," 2024.
40. Capgemini Research Institute, "Upskilling your people for the age of the machine", November 2018.
41. TechTarget, "AI lawsuits explained: Who's getting sued?" May 2024.
42. AWS, "Amazon Q Developer", accessed May 2024.
43. DORA – through such metrics as lead time for changes, deployment frequency, mean time to recovery and change failure rate – measures how well an organization balances speed and stability. The SPACE set of metrics tries to comprehensively assess team dynamics and developer experience. It balances the assessment of technical output with the wellbeing of developers, which traditional metrics fail to do.

Key contributors



Pierre-yves Glever

Executive Vice President, Global head of Cloud & Custom Applications Business Line
pierre-yves.glever@capgemini.com



Serge BACCOU

VP, Cloud and Generative AI, Capgemini Invent
serge.baccou@capgemini.com



Antoine Aymer

Global CTO, Quality Engineering & Testing, Sogeti
antoine.aymer@sogeti.com



Tvishee Kumar

Senior Manager, Capgemini Research Institute
tvishee.kumar@capgemini.com



Stephane GIRARD

CTO, Global Cloud & Custom Applications
stephane.girard@capgemini.com



Anne-Laure THIBAUD (THIEULLENT)

Executive Vice President, Data, AI & Analytics Group Offer Leader
annelaure.thibaud@capgemini.com



Jerome Buvat

Head of the Capgemini Research Institute
jerome.buvat@capgemini.com



Shahul Nath

Manager, Capgemini Research Institute
shahul.nath@capgemini.com



Jiani Zhang

Executive Vice President, Chief Software Officer, Capgemini Engineering
jianiz.zhang@capgemini.com



Jean-Marie Lapeyre

Executive Vice President & Chief Technology and Innovation Officer, Global Automotive Industry
jean-marie.lapeyre@capgemini.com



Amol Khadikar

Research Program Lead, Capgemini Research Institute
amol.khadikar@capgemini.com

The key contributors would like to thank Subrahmanyam KVJ and Harshada Sambare from the Capgemini Research Institute for their contribution to this research.

The key contributors would also like to thank Alex Marandon, David Aline, Julie Spens, Lorna Neville, Mark Oost, Robert Engels, Mark Roberts, Steve Jones, Terry Room, Ashwin Yardi, Lee Smith, Claudia Crummenerl, Paul Gittins, Andy Feinstein, Pascal Brier, Kary BHEEMAIHAH, Gunnar Menzel, Ron Tolido, Nisheeth Srivastava, Bikash Dash, Ajay Mohan, Guillaume Gerard, Rajesh Iyer, Aruna Pattam, Elisa

Farri, Gabriele Rosani, Rajesh K Rangarajan, Julian Fowler, Jayashree Ravichandran, Suresh Arumughom Sarojani, Todd O'Mara, Marc Bauer, Miguel Angel Viedma Lloreda, Donald Hessing, Johanna Sundh, Alexandre Dos-Reis, Erwoan Touche, Marion Lecorbeiller, Fatima-Zahra Houari, Jean-Marie Lapeyre, Sarah Thomas, Mark Buenen, ANTOINE AYMER, Shripal Shah, Udo Lange, David Rutter, Stuart Williams, Supriya Bambawale Yashowardhan Sowale, Andy Vickers, Vincent Biraud, Punam Chavan, Vibha Palekar, Mithun Dutta, Manish Saha and Jaydeep Neogi.



About the Capgemini Research Institute

The Capgemini Research Institute is Capgemini's in-house think tank on all things digital. The Institute publishes research on the impact of digital technologies on large traditional businesses. The team draws on the worldwide network of Capgemini experts and works closely with academic and technology partners. The Institute has dedicated research centers in India, Singapore, the United Kingdom, and the United States. It was recently ranked number one in the world for the quality of its research by independent analysts.

Visit us at www.capgemini.com/researchinstitute/

For more information, please contact:

Global

PIERRE-YVES GLEVER

Head of global Cloud & Custom Applications unit
pierre-yves.glever@capgemini.com

STEPHANE GIRARD

CTO for global Cloud & Custom Applications unit
stephane.girard@capgemini.com

JIANI ZHANG

Chief Software Officer at Capgemini Engineering
jianiz.zhang@capgemini.com

SERGE BACCOU

Vice President, Cloud and Generative AI, Capgemini Invent
serge.baccou@capgemini.com

DAVID ALINE

Lead, Europe Application Managed Services COE
david.aline@capgemini.com

MARK BUENEN

Global Leader, Quality Engineering & Testing, Capgemini Group
mark.buenen@sogeti.com

ANTOINE AYMER

Global CTO, Quality Engineering & Testing, Sogeti
antoine.aymer@sogeti.com

JOAKIM WAHLQVIST

Global CTO, Data & AI, Sogeti
joakim.wahlqvist@sogeti.se

Regional

Americas

SATHEESH SEBASTIAN

Cloud & Custom Applications COE Offer Leader
mail@capgemini.com

North and Central Europe

DANIEL GARSCHAGEN

Director, Innovation and Gen AI North and Central Europe, Germany
daniel.garschagen@capgemini.com

MATTHIAS SONDERMANN

CTO, Cloud and Custom Applications, Germany
matthias.sondermann@capgemini.com

LIZA STURGESS

Director & Chief Architect, UK
lisa.sturgess@capgemini.com

LIZ INCE

Director & Principal Solution Architect, UK
liz.ince@capgemini.com

South and Central Europe

NICOLAS CHANTELOUP

Chief Operating Officer Europe Cluster Business Unity
nicolas.chanteloup@capgemini.com

JEROME FENYO

Head of Business Development, France Application Services
jerome.fenyo@capgemini.com

APAC

MAHESH BHATTAD

Director, Capgemini APAC, Singapore
mahesh.bhattad@capgemini.com

More Capgemini Research Institute Publications



Creative and generative AI



Generative AI in organizations



Generative AI and the evolving role of marketing: A CMO'S playbook:



Eco Digital Era™



The art of software



Open innovation

Subscribe to latest research from The Capgemini Research Institute



Receive copies of our reports by scanning the QR code or visiting

<https://www.capgemini.com/capgemini-research-institute-subscription/>

Capgemini Research Institute

Fields marked with an * are required

First Name *

Last Name *

Email *

By submitting this form, I understand that my data will be processed by Capgemini as indicated above and described in the [Terms of use](#).*

Submit



NOTES



NOTES



About Capgemini

Capgemini is a global business and technology transformation partner, helping organizations to accelerate their dual transition to a digital and sustainable world, while creating tangible impact for enterprises and society. It is a responsible and diverse group of 340,000 team members in more than 50 countries. With its strong over 55-year heritage, Capgemini is trusted by its clients to unlock the value of technology to address the entire breadth of their business needs. It delivers end-to-end services and solutions leveraging strengths from strategy and design to engineering, all fueled by its market leading capabilities in AI, cloud and data, combined with its deep industry expertise and partner ecosystem. The Group reported 2023 global revenues of €22.5 billion.

Get the Future You Want | www.capgemini.com