E-book

Microsoft

Redefining the Developer Experience with AI and DevOps

How integrating AI-enabled tools into DevOps improves developer satisfaction

Table of contents

Introduction

The intersection of AI and DevOps

In this era where businesses must innovate both to survive and thrive, software development teams are increasingly vital. While this has elevated their status, it's also heightened their workload and need to learn new skills—especially as coding has become more intertwined with security and IT management workflows. Without proper tools and support, this pressure can lead to decreased productivity, creativity, and job satisfaction, resulting in high turnover rates.

An astounding 92% of developers are already using AI coding tools, with the majority using them for personal projects in addition to their jobs. However, as with any transformative new technology, many organizations are still exploring how to get the most value from it.

In this e-book, we'll provide guidance and research insights on how developer satisfaction is positively impacted by the adoption of AI-enabled tools and DevOps—two areas that are now inextricably linked. We'll also provide an overview of specific Microsoft and GitHub AI and DevOps tools, and resources for how to combine them to improve developer experiences and outcomes in your organization.

This e-book is for you if:

You lead development teams and are tasked with technology and tools decisions for your organization.

You need to invest in technology and processes that help developers be more productive, creative, and satisfied.

You want to learn about popular solutions and how other companies are using AI and DevOps together.

Section 01

Why 92% of developers are using AI coding tools

A GitHub survey[1] that focused on AI's impact on the developer experience revealed that 92% of devs are already using AI coding tools—and the majority of them (67%) are using them both at work and at home. While hype certainly played a role in this sudden widespread adoption, the research reveals a more compelling reason for the appeal: 70% of developers said that using AI has improved their work.

Specifically, they reported improved code quality, faster outputs, and fewer production-level incidents with AI. However, this new ability to produce code faster poses a challenge for dev team leaders: what to do with this new productivity.

There's been an industry-wide shift away from measuring dev productivity by the volume of code produced, but a third of devs report it's still the main metric their organization uses. They're feeling increasingly dissatisfied by that and would like to see their managers adopt new ways of measuring productivity that take quality and security into account—which are elements that drive satisfaction for devs as well as better business outcomes.

Business leaders are also embracing AI[2]

Leaders who say their companies are already using AI: **71%**

Organizations are realizing a return on AI investments within **14** MONTHS

For every dollar invested in AI, the average return is **$3.50**

However

Leaders who say their biggest barrier to implementing and scaling AI is a lack of skilled workers: **52%**

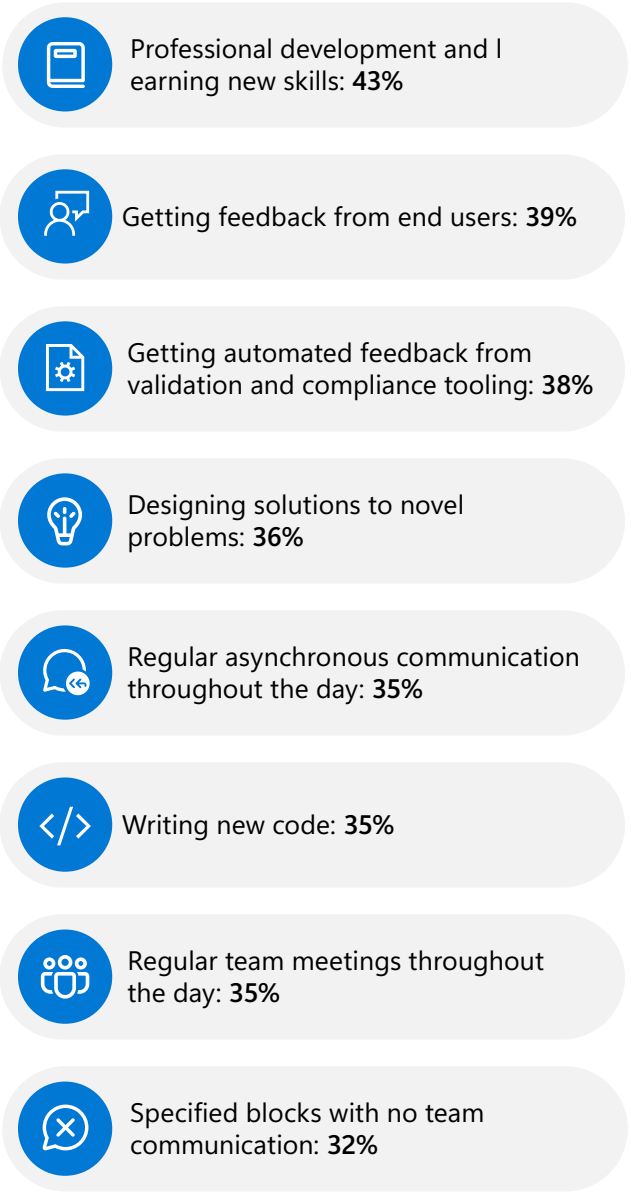
Enhancing creativity, problem-solving, and collaboration

The astonishingly high rate at which devs have embraced AI coding tools in their personal projects reveals a critical truth about what they're passionate about: creativity. They'd like to use these tools to bring more of their creativity into the workplace, too.

A major factor in devs' enthusiasm for AI coding tools is that they make it easy to explore new approaches and diverse coding styles in the context of real projects. In the GitHub survey, developers reported that this opened new avenues for creative problem-solving, leading them to design more innovative and effective solutions.

Additionally, 80% of devs said they believe that AI coding tools will enhance team collaboration. GitHub Copilot, in particular, makes standardizing coding practices across teams more efficient by removing a barrier to collaboration between developers with different backgrounds. Adopting DevOps and a robust integrated developer environment, which we'll discuss in the next chapters, also gives a major boost to collaboration.

The top tasks developers say positively impact their workdays



Increasing satisfaction and reducing burnout

It's clear that the impact of AI on developer satisfaction is profound. The GitHub survey results indicate that learning new skills (43%), receiving feedback from end users (39%), and getting automated feedback from validation and compliance tooling (38%) are the top three contributors to a positive workday. AI-enabled tools—GitHub Copilot, specifically—help them focus more on those areas.

The survey also indicates that these tools help reduce non-critical mental effort. With both mind and time being limited resources, 41% of devs agreed these tools can help avoid burnout. That said, managers play a critical role in improving satisfaction by reallocating time saved on coding to focus on quality, security, and innovation.

More productive

Developers who find their work engaging feel they're **30% more productive** compared to those who find their work boring.[3]

Securing code from the start

We'll go into more depth on security in the next chapters, but it's important to note here that GitHub Copilot is designed to help block insecure coding patterns and spot vulnerabilities proactively during all stages of development. However, it's also vital to build dedicated code scanning, secret scanning, and dependency scanning right into your development workflow, which you can do with GitHub Advanced Security.

As always, developers must remain diligent in code reviews to maintain a security-innovation balance, but these security features are particularly valuable for dev teams that still do most of their security testing in later stages.

activity != productivity

Implementing the research-backed SPACE framework is a proven way for managers to evolve their teams to focus on outcomes, rather than output, to improve dev productivity and satisfaction.

It's called "Copilot," not "Autopilot," for a reason

GitHub Copilot makes coding more efficient, but it's not a magic solution that can write perfect code for you. It assists you and complements your skills but requires human judgment and supervision.

Example of a vulnerable pattern flagged by GitHub Copilot:

```
script.src = "https://maps.azureapis.com/maps/api/js?key=[REDACTED]"
script.async = true;
script.defer = true;
document.body.appendChild(script)
```

Section 02

Focusing on deep work with AI and DevOps

The term deep work, coined by the writer Cal Newport, refers to the state of focused and uninterrupted concentration that allows you to perform cognitively demanding tasks effectively and efficiently. Deep work is essential for developers who need to solve complex problems, create innovative solutions, and learn new skills. Yet it's often hindered by distractions and inefficiencies in the modern software development process.

This is where DevOps presents a powerful opportunity to improve the developer experience. It increases confidence in the quality and security of applications and fosters coordination and collaboration between often-siloed areas of software development, IT operations, quality assurance (QA), and security. And, especially when integrated with a solid dev tool stack, it reduces toil on repetitive tasks, allowing devs to spend more time doing what they love: being creative.

Developer experience takes into account:

- **How simple and fast it is for a developer to implement a change on a codebase—or be productive.**
- **How frictionless it is to move from idea through production to impact.**
- **How positively or negatively the work environment, workflows, and tools affect developer satisfaction.**

But what *is* DevOps?

There's spirited debate among developers about whether DevOps is a methodology, a practice, or a philosophy. For the purposes of this e-book, we'll define it as:

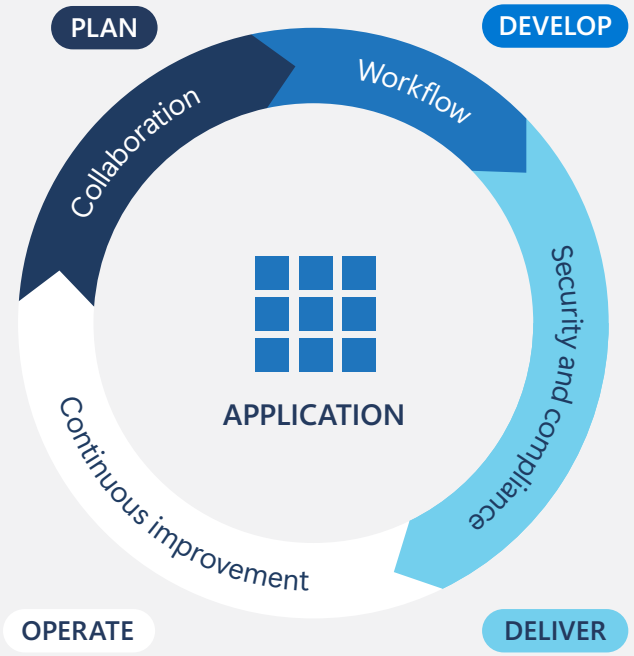
a holistic approach that brings together people, processes, and technology to deliver value to customers in a continuous and efficient way.

What DevOps *isn't*—deconstructing two common myths:

1. **Having DevOps tools equals doing DevOps.**
   Tooling is important, but it's not DevOps without culture and processes.
2. **DevOps is agile development.**
   While there's overlap between DevOps and agile development, DevOps also focuses on operations.

Adopting DevOps

While the majority of developers have already begun adopting AI-enabled tools on their own, adopting DevOps needs to happen at the organizational level. This involves determining the organization's needs, current state of adoption, and goals.



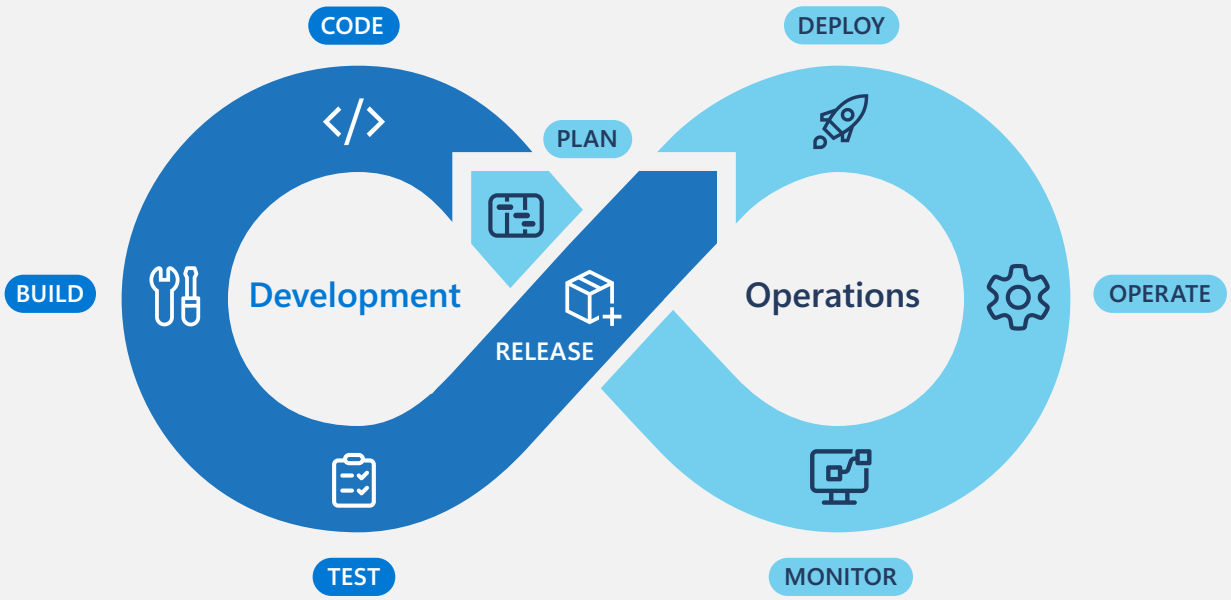
Tip:

You'll find resources at the end of this e-book to help you plan and implement DevOps at your org, as well as to combine powerful tools such as Azure DevOps, GitHub Copilot, Visual Studio, Visual Studio Code, and GitHub Advanced Security for Azure DevOps.

A quick overview of the DevOps adoption process

1. **Understand DevOps.** To adopt DevOps, first grasp its core principles, including continuous integration, automated testing, and infrastructure as code.
2. **Shift the culture.** Promote a collaborative culture, break down silos, and foster open communication between development and operations teams.
3. **Choose the right tools.** These include source control systems, security services, and configuration management and monitoring tools. You can start small and add tools as your team gains experience.
4. **Automate everything.** Implement automation to streamline build processes, testing, deployments, and infrastructure management to improve efficiency and reduce errors.
5. **Secure everything.** Help better protect your code and data by building security into your entire process—not just at a single stage.
6. **Create CI/CD pipelines.** Use continuous integration and continuous delivery (CI/CD) to integrate changes and deliver them to production automatically. This helps catch issues early and makes deployments more predictable.
7. **Monitor and improve.** Continuously monitor and optimize DevOps processes by reviewing performance metrics, conducting postmortems, and seeking team feedback.

THE DEVOPS PROCESS



Section 03

Integrating AI-enabled tools and security into DevOps

Now let's move on to what, for many folks, is the fun stuff: new tools. Incorporating Microsoft and GitHub products into your DevOps practice is a great way to provide developers with the tech stack and processes they need to improve the quality, resilience, security, and impact of their applications. There are over 1,000 apps and services—many of which are community-built—that can be used with Azure and GitHub, but here we'll focus on just AI and security.

Note that the details in the following chart are far from exhaustive, so if you don't see something you're looking for, please refer to the product pages for more information.

Customer spotlights

Novo Nordisk accelerates life-changing research by boosting collaboration and security using GitHub with Azure DevOps.

Read more >

Norwegian insurance leader Gjensidige builds on GitHub and Azure to put security front-and-center in a new application platform.

Read more >

Microsoft and GitHub AI and DevOps tools

Product	Description	Often used with	Commonly used languages and frameworks
Azure AI services	A collection of cloud-based services and tools for building AI-enabled applications with ease and confidence. Includes prebuilt and custom models, cognitive services, machine learning, and conversational AI.	GitHub, GitHub Copilot, Microsoft Dev Box, Visual Studio, Visual Studio Code, and Azure DevOps.	.NET, C#, Java, JavaScript, Python, SQL Server
Azure Deployment Environments	Self-service, project-based templates to securely deploy consistent environments for any stage of development. Empowers innovation through collaboration, experimentation, and propagation of best practices.	GitHub, GitHub Copilot, Azure Resource Manager templates, Microsoft Dev Box, and Terraform.	
Azure DevOps	A comprehensive suite of services and integrations that enable you to plan, develop, test, deploy, and monitor software projects of any size, complexity, and platform. Supports agile methodologies, CI/CD pipelines, cross-platform development, QA, security, and compliance.	GitHub, GitHub Advanced Security for Azure DevOps, Microsoft Dev Box, Visual Studio, Visual Studio Code, and Azure AI services.	.NET, Docker, Java, Jenkins, Kubernetes, React Native, SQL Server, Xamarin
GitHub Advanced Security	A security solution that provides code scanning, secret scanning, and dependency scanning to help you secure your application development.	GitHub, GitHub Copilot, Microsoft Defender for Cloud, and Microsoft Power BI.	
GitHub Advanced Security for Azure DevOps	A security solution that brings the best of GitHub security to Azure DevOps. It provides code scanning, secret scanning, and dependency scanning to help you secure your application development.	GitHub, GitHub Copilot, Visual Studio, Visual Studio Code, Azure DevOps, Microsoft Defender for Cloud. Also used with Azure Policy to enforce security policies and compliance across projects.	
GitHub Codespaces	A cloud-based development environment that lets you code from anywhere, on any device. It provides a fully configured, containerized, and customizable environment that is ready to go in minutes. It also syncs with your GitHub projects and settings.	GitHub, GitHub Copilot, GitHub Advanced Security for Azure DevOps, Visual Studio Code, Azure DevOps, and Azure AI services.	Angular, Go, JavaScript, Python, React, Ruby, TypeScript
GitHub Copilot	An AI pair programmer that suggests code as you type, based on your context and comments. It can generate whole functions or snippets of code, write tests, handle multiple languages and frameworks, and learn from your code and feedback.	Visual Studio Code, Visual Studio 2022, GitHub Codespaces, Azure DevOps, and Azure AI services.	
Github Enterprise	A development platform with end-to-end security, the latest in automation, and tools to help your team do their best work yet.	GitHub, GitHub Copilot, Azure DevOps, and Azure AI services.	Go, Hashicorp Configuration Language, Java, JavaScript, Lua, Python, TypeScript, Rust
Microsoft Dev Box	A standardized development environment that provides a consistent and reliable way to develop, test, and debug software. It includes preconfigured tools, templates, and settings that suit your project and preferences.	GitHub, GitHub Copilot, Visual Studio, Visual Studio Code, and Azure DevOps.	.NET, C#, Java, JavaScript, Python, SQL Server
Visual Studio	An integrated development environment (IDE) that offers a rich set of features and tools for code editing, debugging, testing, collaboration, and deployment.	GitHub, GitHub Copilot, GitHub Advanced Security for Azure DevOps, Microsoft Dev Box, Azure DevOps, and Azure AI services.	Docker, Kubernetes, React Native, SQL Server, Xamarin
Visual Studio Code	A lightweight and versatile code editor that offers a fast and smooth development experience. Supports various languages and frameworks, with many extensions available to enhance its functionality.	GitHub, GitHub Copilot, GitHub Advanced Security for Azure DevOps, Microsoft Dev Box, Azure DevOps, and Azure AI services.	Django, Docker, Flask, Kubernetes, PyGame

Next steps

Planning your developer productivity solution

The interoperability of GitHub Copilot, Azure DevOps, and Visual Studio significantly enriches the developer's world, making the software development process more intuitive, efficient, and enjoyable. This combination not only enhances the individual capabilities of each tool but also creates a cohesive environment where developers can thrive.

By streamlining workflows and fostering a space for innovation, developers find themselves more satisfied and engaged, leading to higher productivity, improved business outcomes, and increased retention. The continued evolution of AI and DevOps promises a future where developers can seamlessly turn their visions into reality. This nexus of technology and creativity is at the heart of the Microsoft and GitHub developer stack, providing secure, flexible solutions that empower every developer and developer team.

Get started:

- Discover how to use GitHub and Azure together.
- Take the Introduction to GitHub Copilot learning module.
- Calculate your organization's Developer Velocity score and get guidance to boost business performance.

1 Shani, Inbal. "Survey Reveals AI's Impact on the Developer Experience." The GitHub Blog, September 29, 2023. https://github.blog/2023-06-13-survey-reveals-ais-impact-on-the-developer-experience/.

2 IDC infographic, sponsored by Microsoft, The Business Opportunity of AI: How leading organizations are using AI to drive impact across every industry and addressing barriers such as AI governance, upskilling, and cost, US51315823, 2023.

3 Forsgren, N., Kalliamvakou, E., Noda, A., Greiler, M., Houck, B., Storey, M. 2024. DevEx in Action: A study of its tangible impacts. acmqueue 19, https://dl.acm.org/doi/10.1145/3639443.

Microsoft