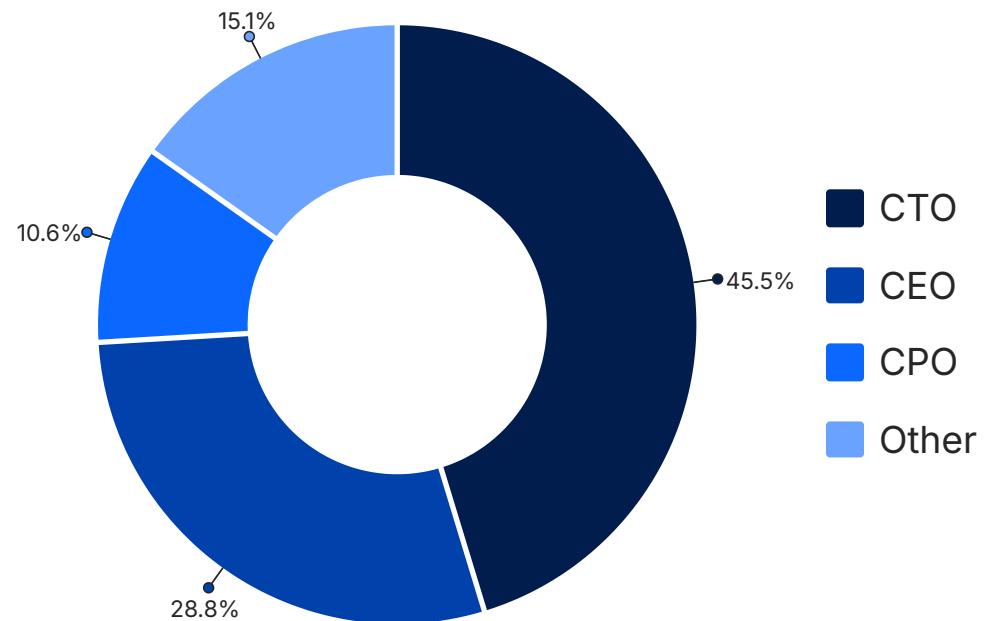


# Exploring Challenges and Opportunities in Software Engineering: a Thunders study

*March, 2025*

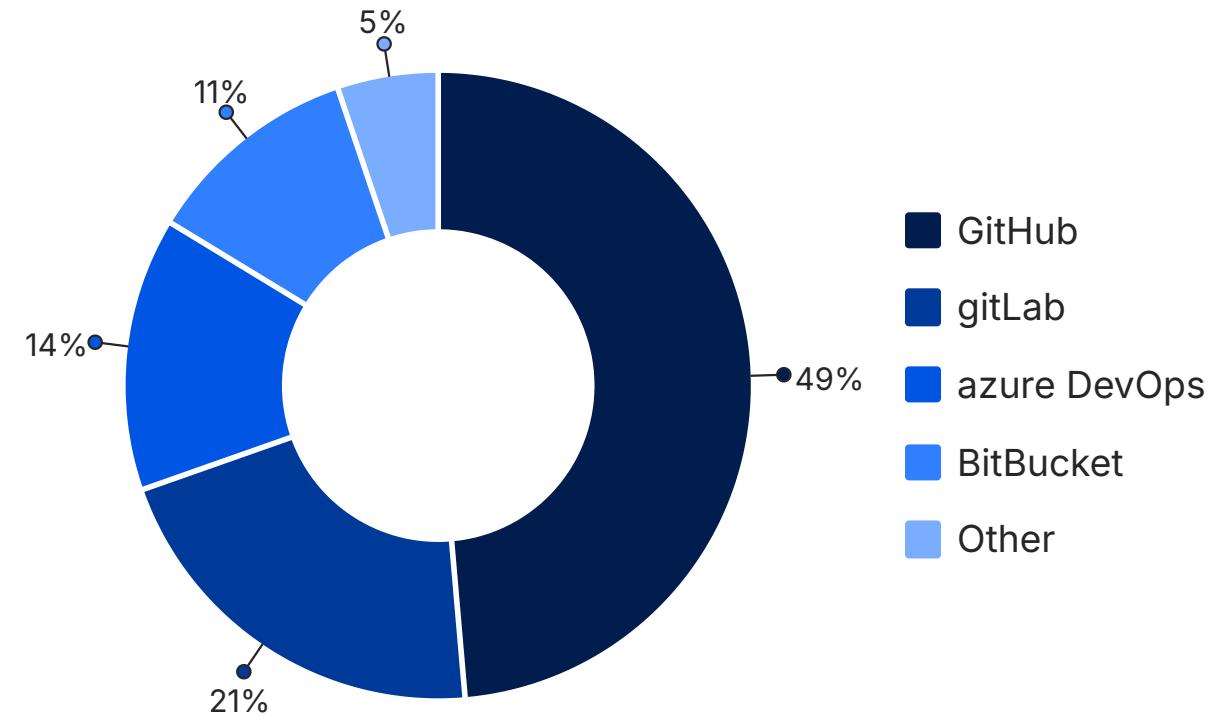
# Leadership Composition:

CTOs dominate the respondent pool, showing that **tech decision-makers drive automation priorities** in most companies.



# Version control systems usage:

GitHub dominates version control (49%), but CI/CD choices are fragmented. This reflects teams at different maturity stages making different infrastructure bets.

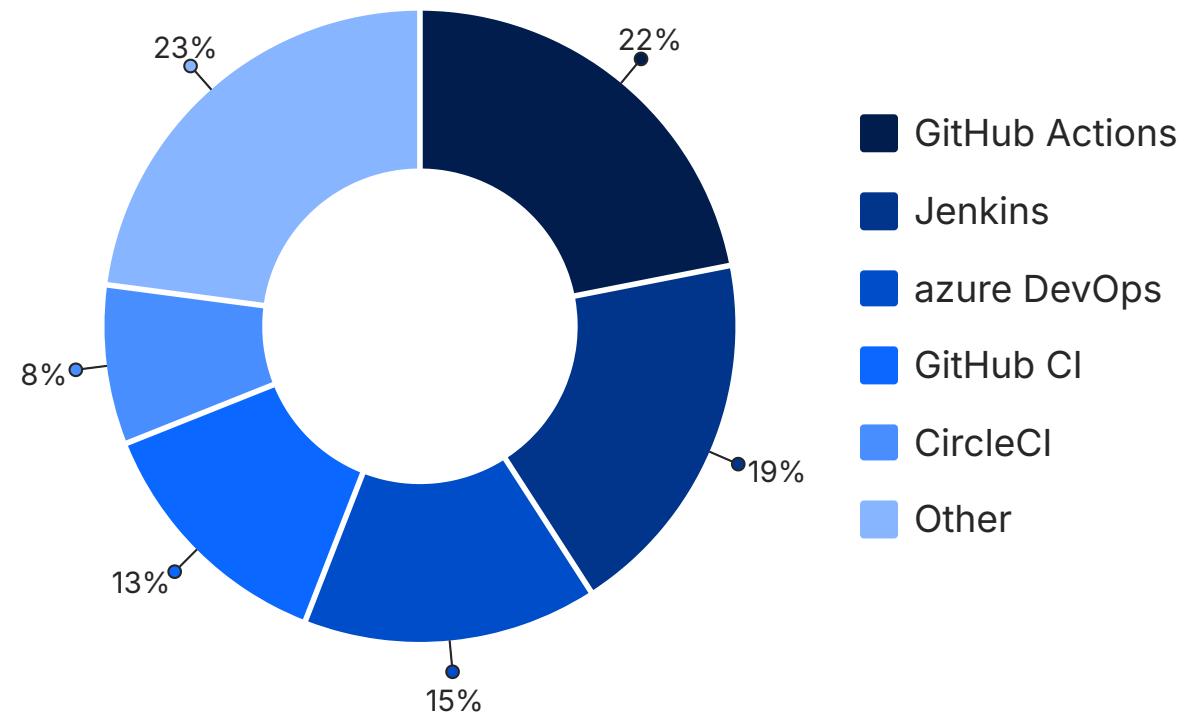


# CI/CD Tool Adoption for Development Pipelines:

**GitHub dominates CI/CD adoption, but the landscape is fragmented.**

Many teams experiment with Jenkins, Azure DevOps, or niche tools, reflecting diverse approaches to building and shipping software pipelines.

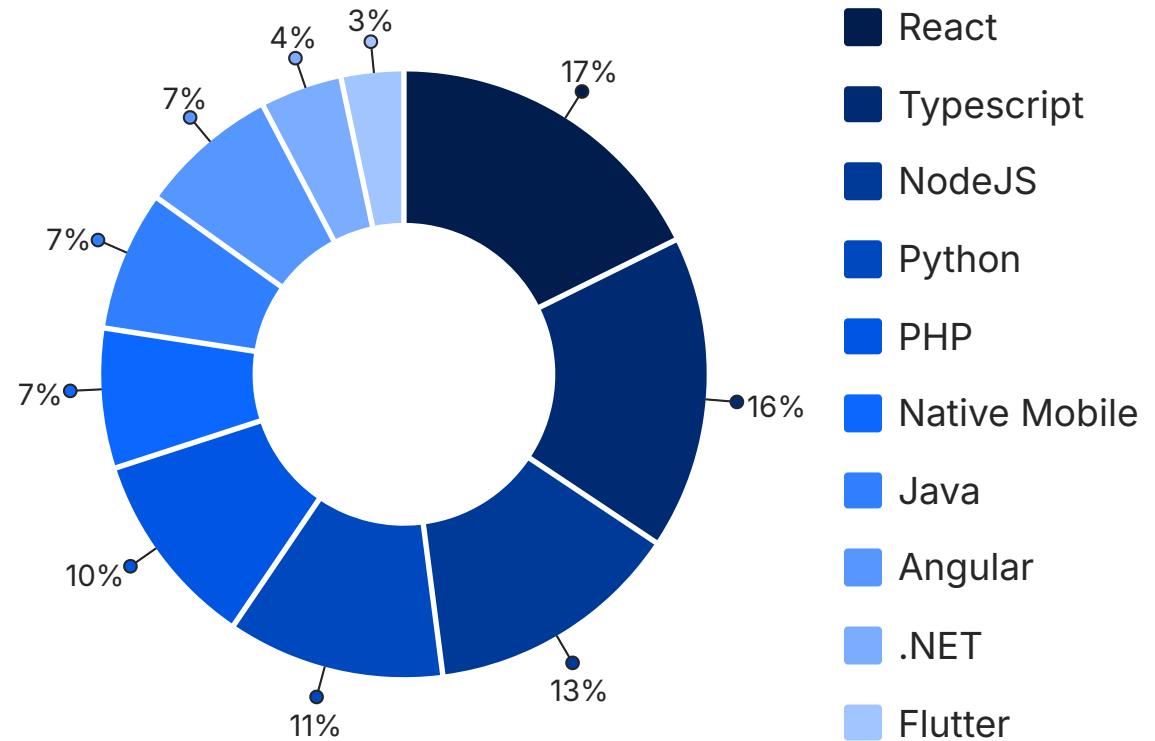
*Even within GitHub, teams split between Actions and CI: no single approach fits all.*



# Programming Languages & Framework Usage:

Front-end frameworks dominate the stack, with **React**, **TypeScript**, and **NodeJS** leading adoption.

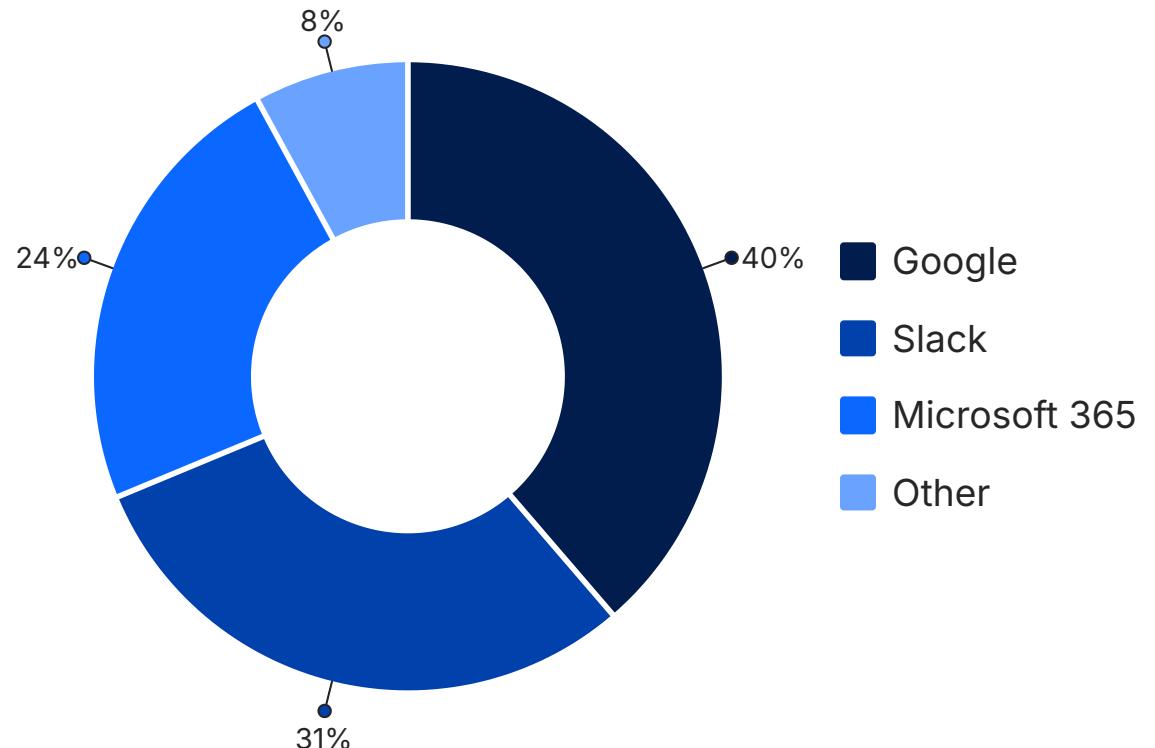
**The low presence** of .NET and Flutter **reflects the youth and cloud-native focus** of these companies.



# Collaboration Suite Usage:

Teams overwhelmingly rely on mainstream tools like **Google Workspace, Slack, and Microsoft 365.**

Alternatives exist but remain niche, *showing that collaboration standards are firmly established.*

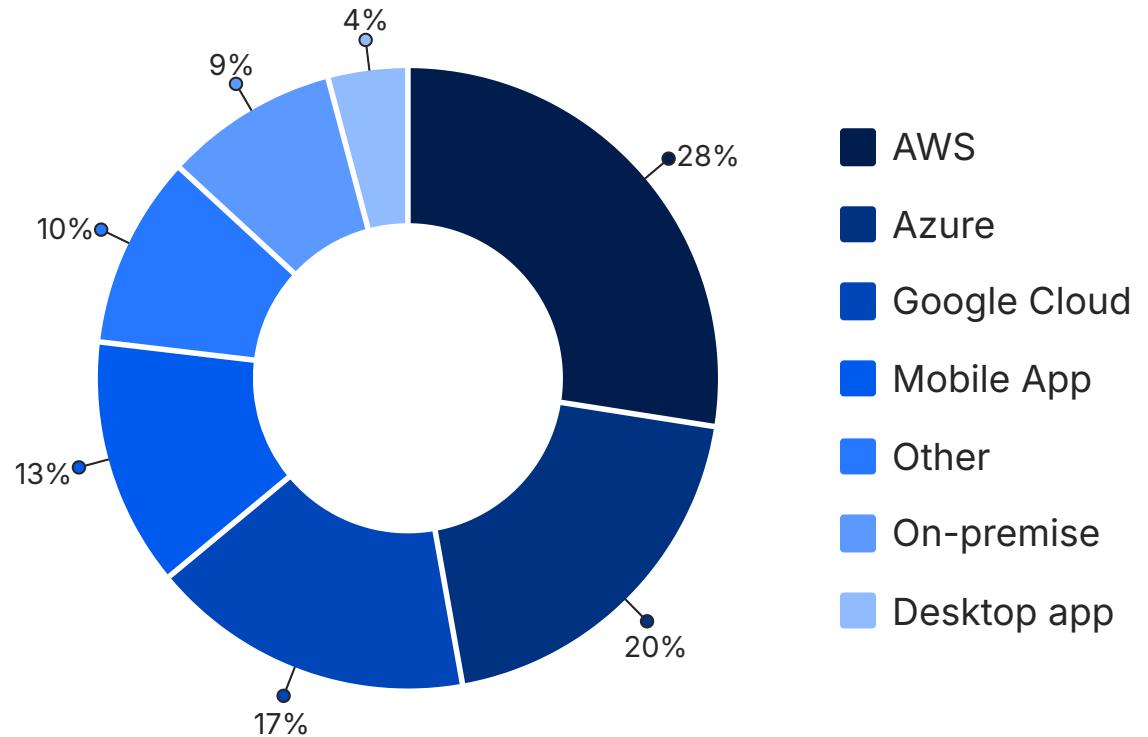


# Deployment Platform Distribution:

The majority of teams are **cloud-first**,  
**led by AWS**, with **significant mobile deployment**.

Desktop apps are nearly extinct, but on-premise still matters for companies with specific infrastructure needs.

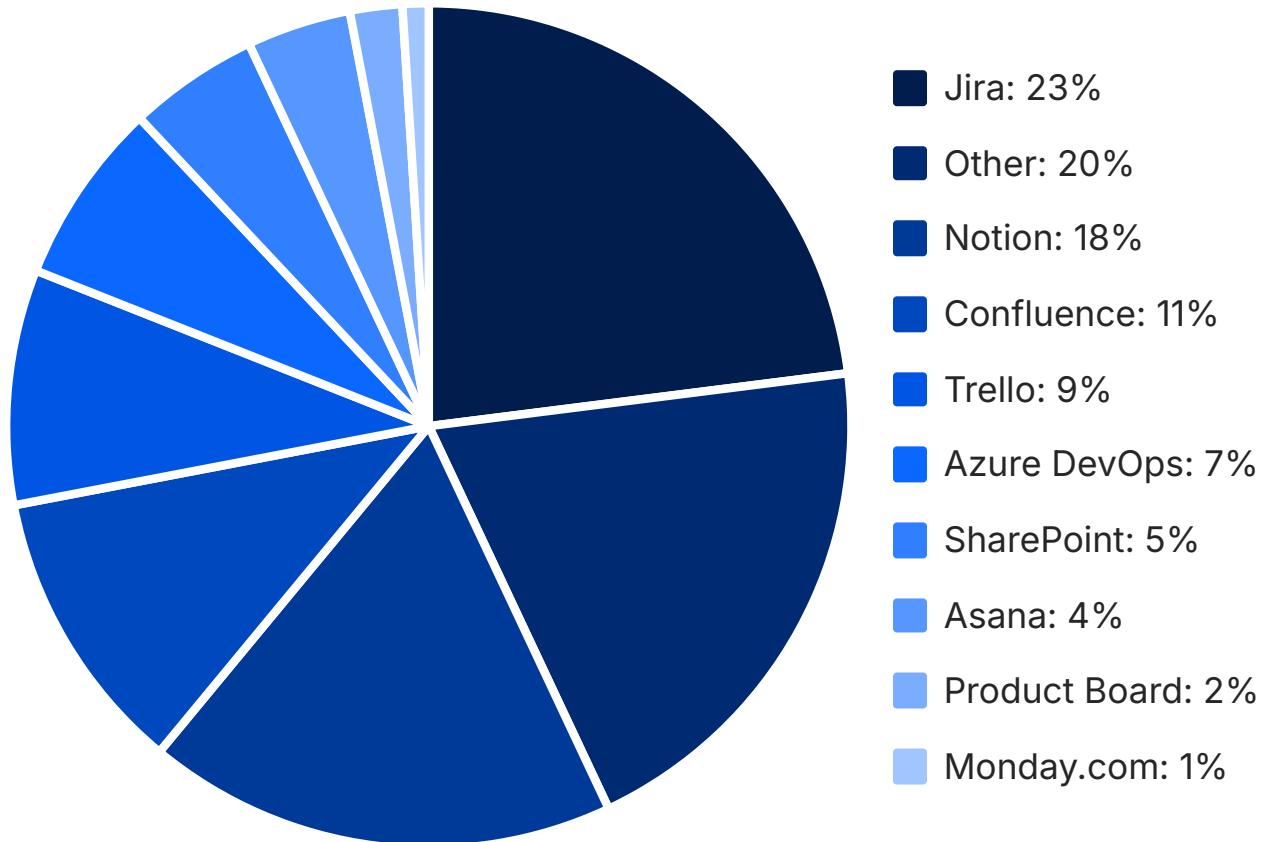
*These choices show a shift toward fast, scalable, and user-facing products over legacy systems.*



# Project & Knowledge Management:

**Jira and Notion are the most widely adopted tools, but 20% of teams rely on alternative solutions, highlighting a fragmented landscape.**

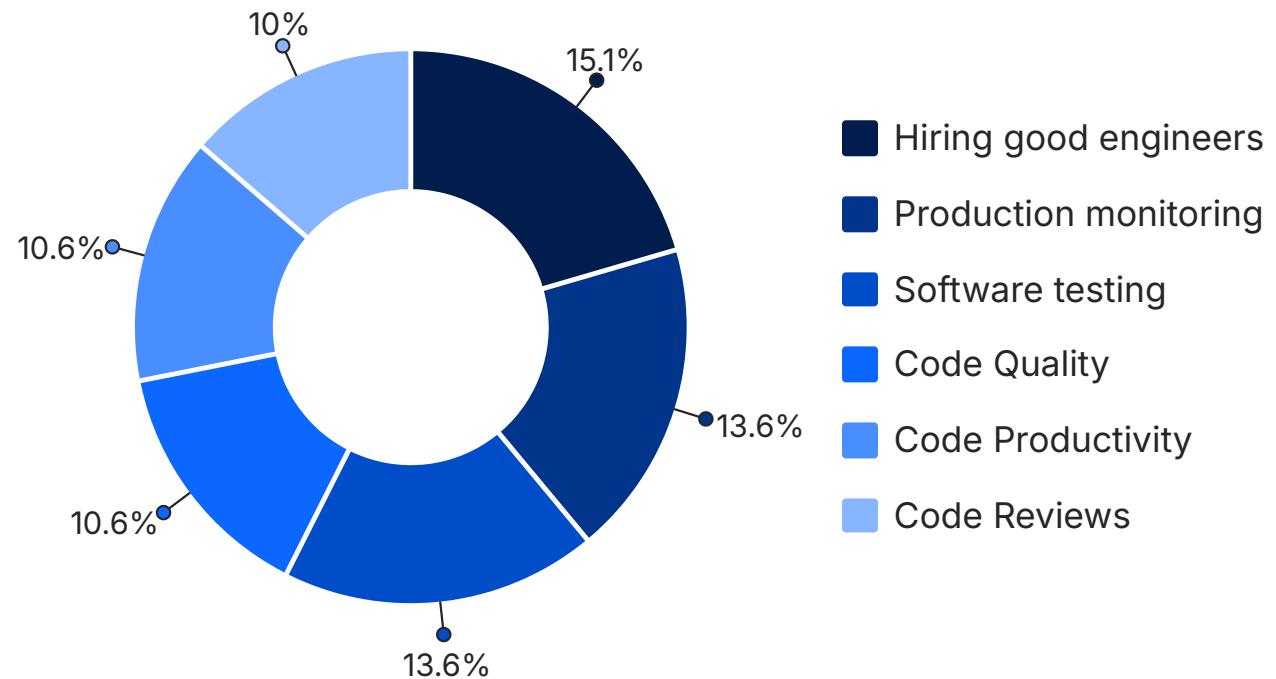
*Teams tailor workflows to their specific needs, which can make cross-team alignment and process consistency more challenging.*



# Key Challenges in Software Engineering:

**Hiring and retaining skilled engineers remains the top challenge** for most teams, followed closely **by production monitoring and software testing**.

While basic operational tasks are generally well-managed, quality assurance, monitoring, and documentation continue to demand consistent attention.

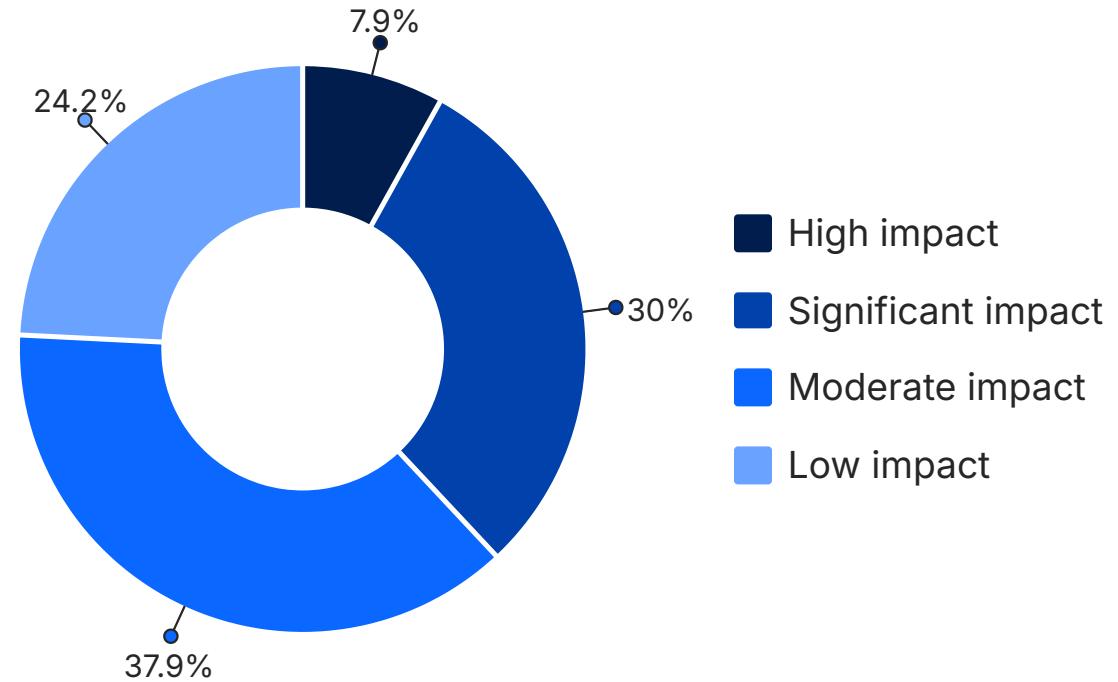


# Impact of Challenges on Success:

Engineering challenges affect success across the board:

Three-quarters of teams report moderate to high impact, and over a third see their effect as significant or high.

**This underscores that issues like talent acquisition, QA, and monitoring directly shape product outcomes.**

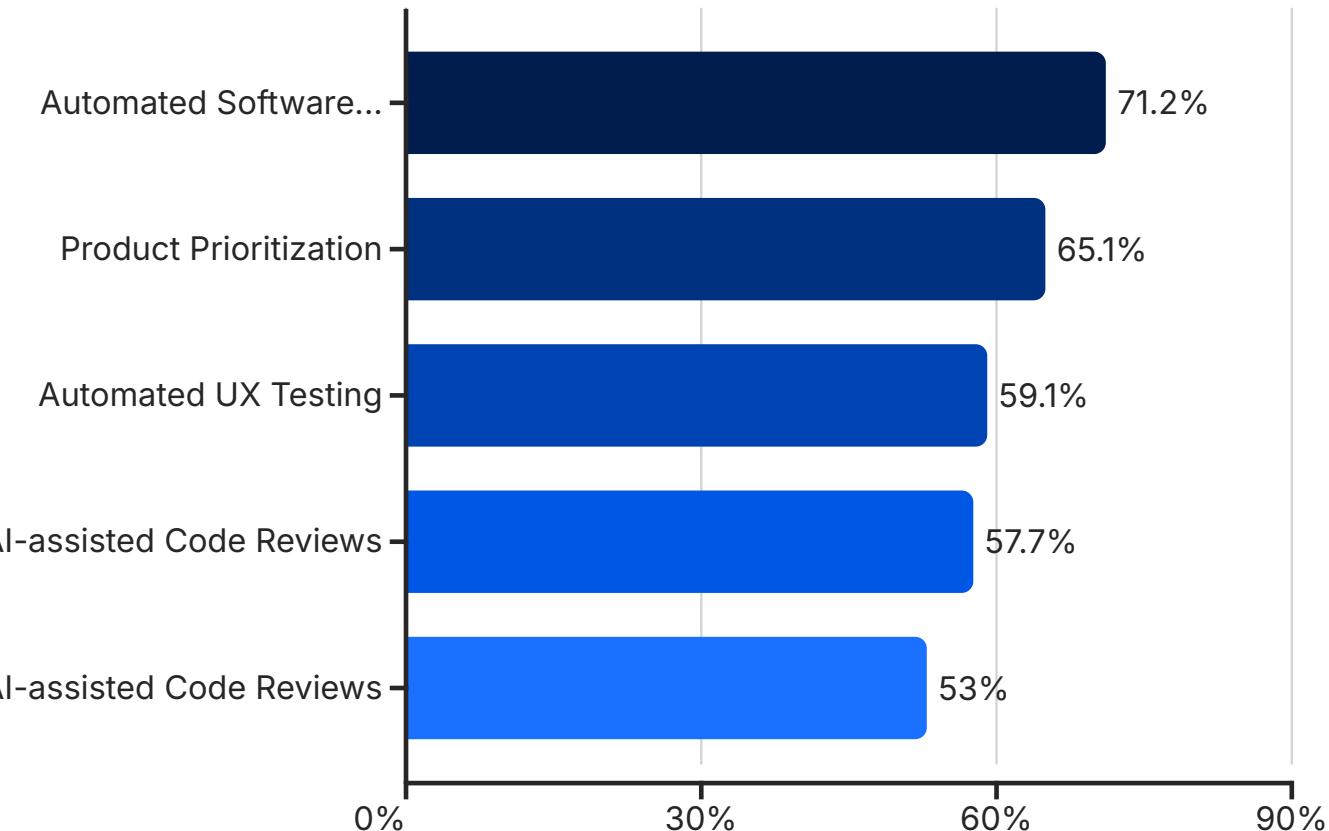


# Value of AI & Automation in Software Engineering:

**AI & automation are broadly valued across workflows.**

Testing leads clearly, while product prioritization and code-focused AI tools also show strong support. Low skepticism suggests these capabilities are becoming expected, not experimental.

*The consensus is shifting **from whether** to adopt AI and automation **to where** they create the most immediate value.*

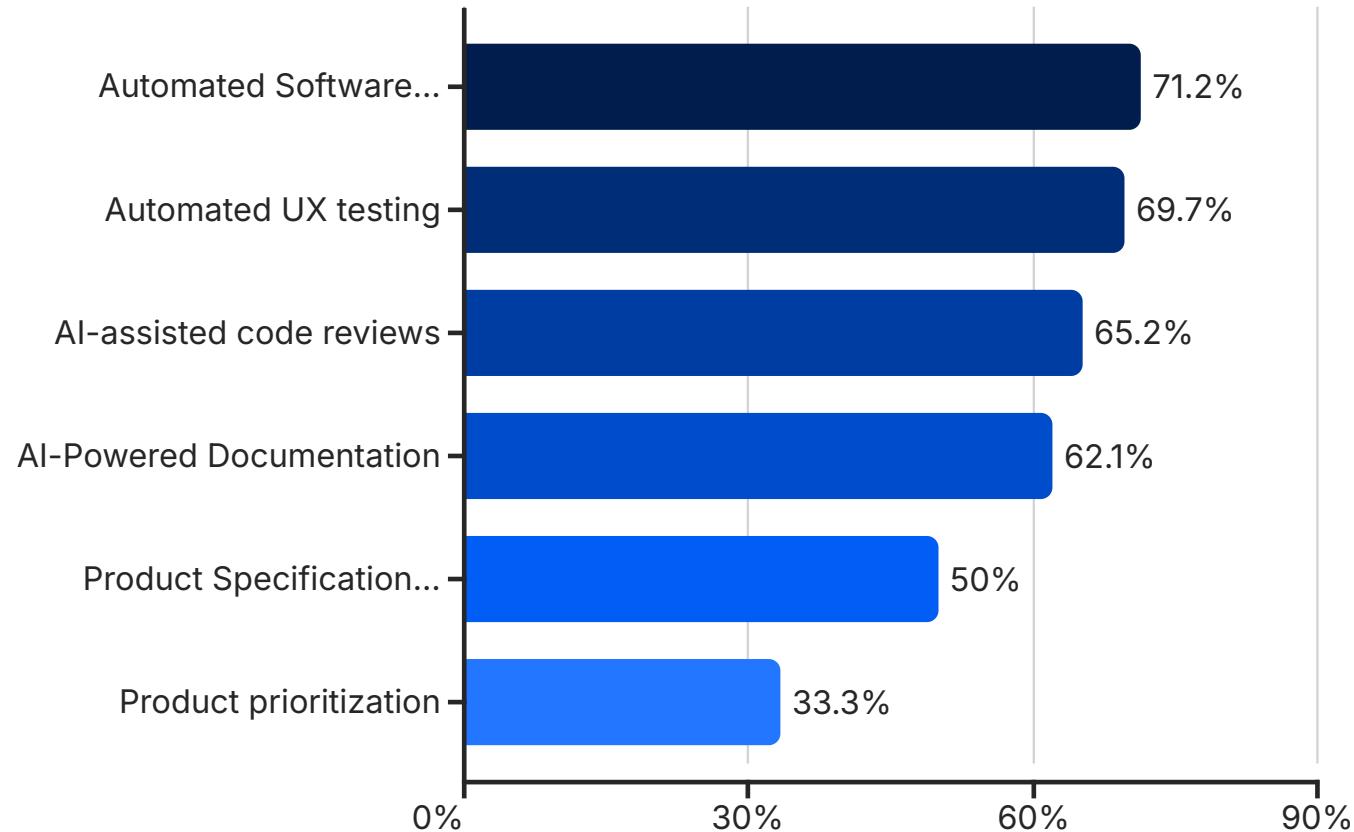


# Value of Development Practices in Software Engineering:

**Automated software and UX testing lead in perceived value.**

AI-assisted code reviews and documentation are gaining traction, while product planning, especially prioritization, shows more mixed adoption.

*Teams prioritize practices that directly improve code quality and speed delivery.*



# Key Findings:

## Highest value:

### Automated software testing

71.2% of respondents rated **automated software** testing as either "extremely" or "very" **valuable to their development process.**

## Lowest Value:

### Product Prioritization

Only **33.3%** of respondents consider **product prioritization** either "extremely" or "valuable", **making it the lowest-rated practice in our survey.**

## Most Consistent Reception:

### AI-assisted code reviews

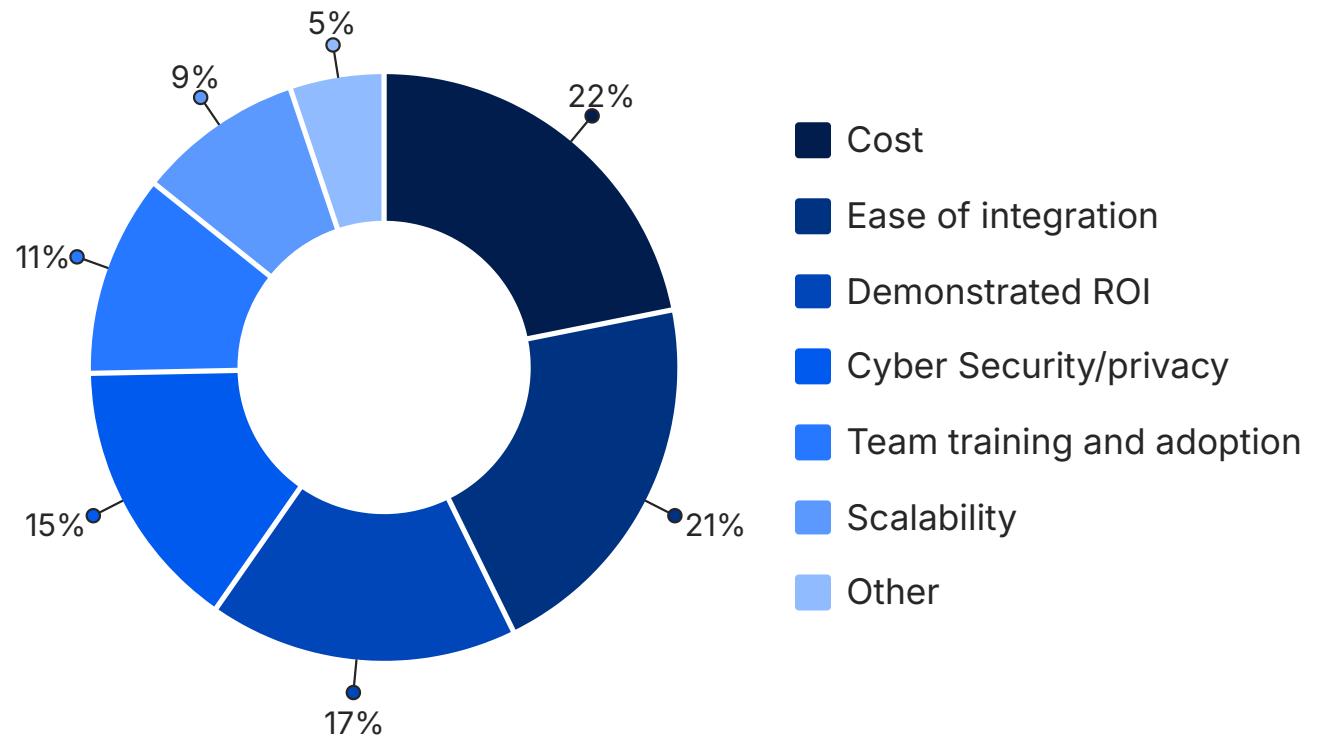
**AI-assisted code reviews** received the most consistent positive submission, with **83%** of respondents acknowledging their value.

# Factors Influencing AI Adoption:

**Cost and ease of integration are the top considerations** when adopting AI solutions.

Security, ROI, and team readiness also *influence* decisions, while scalability and other factors play a smaller role.

*Practical and operational concerns outweigh purely technical or strategic factors in AI adoption choices.*



# Future Role of AI in Software Engineering:

**Most respondents expect AI to boost automation and productivity, streamlining routine tasks and improving code quality.**

A significant portion foresee evolving engineering roles and deeper integration of AI into development tools.

*AI is seen as both a force multiplier for productivity and a driver of change in how teams work.*

