

The SPACE of

Developer Productivity

There's more to it than you think



Hi, I'm Karthik.

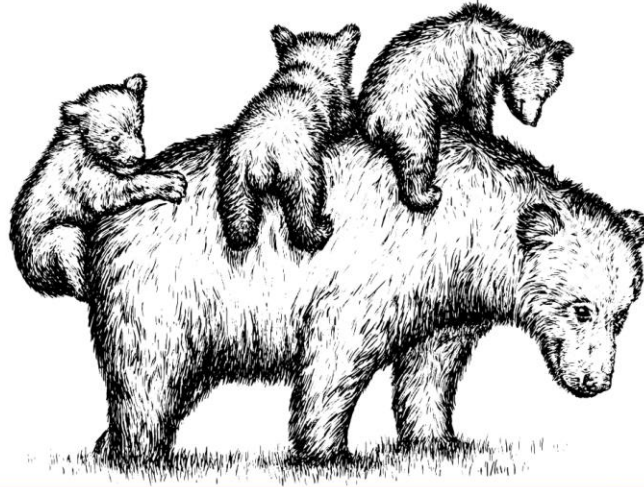
Karthik Rameshkumar, *Director - Solutions Engineering*
linkedin.com/in/karthik-rameshkumar



Karthik Rameshkumar
Director @ GitHub | Leading
APAC West Solutions Engin...



Getting the wrong idea from that conference talk you attended



Solving Imaginary Scaling Issues

At Scale

O RLY?

@ThePracticalDev

Why developer productivity matters

Learning how to define, measure, and predict developer productivity can help organizations:



Make higher-quality software



Produce software more efficiently



Make smarter planning decisions



Improve the well-being and satisfaction of developers



Facilitate team collaboration and success



On behalf of all EMs out there

This is how it feels



Myths and misconceptions about developer productivity

You can't improve what you don't understand. Here are the common myths and misconceptions around developer productivity:

01

Productivity is all about developer activity

02

Productivity is only about individual performance

03

One productivity metric can tell you everything

04

Productivity measures are only useful for managers

05

Productivity comes down to engineering systems and developer tools



SPACE: A framework for understanding developer productivity

The SPACE framework offers a new way of thinking about developer productivity.



Satisfaction and well-being



Performance



Activity



Communication and collaboration



Efficiency and flow

S

Satisfaction and well-being

Satisfaction

How fulfilled developers feel with their work, team, tools, or culture

Well-being

How healthy and happy developers are

Metrics:

- Employee satisfaction
- Developer efficacy
- Engagement
- Burnout



P

Performance

Performance

The outcome of a system or process.
It's hard to quantify performance
because there are so many variables.

Metrics:

Quality of code

- Reliability
- Absence of bugs
- Ongoing service health

Impact of code

- Customer satisfaction
- Customer adoption and retention
- Feature usage
- Cost reduction



A

Activity

Activity

the number of actions or outputs completed in the course of performing work

Metrics:

- Design and coding
- Continuous integration and deployment
- Operational activity
- Intractable activities



C

Communication and collaboration

Communication and collaboration capture how people and teams communicate and work together.

Metrics:

- Discoverability of documentation and expertise
- How quickly work is integrated
- Quality of reviews of work contributed by team members
- Network metrics that show who is connected to whom and how



E

Efficiency and flow

Efficiency and flow

refer to how well developers and teams can make progress on their work or complete it without interruptions or delays.

Metrics:

- Number of handoffs in a process; number of handoffs across different teams in a process
- Perceived ability to stay in flow and complete work
- Interruptions
- Time measures through a system



Review

How to use the framework

Measuring metrics across at least three different dimensions of the framework.

Balance your metrics:

- Consider your organization
- Review multiple metrics

Remember

Metrics signal what a company values. The more varied the metrics, the more balanced your picture of overall developer productivity.



Using **SPACE**

Copilot meets SPACE

How GitHub Copilot supports productivity



Developers distrust productivity tools



Anxiety-inducing



Often used against
devs




Inaccurate/unfair
representation of
devs' work



Feel like spying





Talk is Cheap.
Show me the
Productivity!

Your application is a special snowflake



Expert

Excuses for Not Writing Unit Tests

ORLY?

@ThePracticalDev

Unit Tests



GITHUB COPILOT

(date, value, currency) but it is currently (value, currency, date).
3. The format string used in `strftime()` is incorrect. It should be "%Y-%m-%d" instead of "%d-%m-%Y".

Here's the corrected code:

```
def parse_expenses(expenses_string):  
    """Parse the list of expenses and return  
    list of triples (date, value, currency).  
    Ignore lines starting with #.  
    Parse the date using datetime.  
    Example expenses_string:  
        2023-01-02 -34.01 USD  
        2023-01-03 2.59 DKK  
        2023-01-03 -2.72 EUR  
    """  
    expenses = []  
  
    for line in expenses_string.splitlines():  
        if line.startswith("#"):  
            continue  
        date, value, currency = line.split(",")  
        expenses.append((datetime.datetime.strptime(date, "%Y-%m-%d"),  
                        float(value),  
                        currency))
```

How can I use the `parse_expenses` function in my code?

Ask Copilot a question or type '/' for topics

PrimeTestCase.py M • utils.py M • parse_expenses.py M X

github-copilot-examples > python > parse_expenses.py > parse_expenses

```
1 import datetime  
2  
3 def parse_expenses(expenses_string):  
4     """Parse the list of expenses and return the  
5     list of triples (date, value, currency).  
6     Ignore lines starting with #.  
7     Parse the date using datetime.  
8     Example expenses_string:  
9     2023-01-02 -34.01 USD  
10    2023-01-03 2.59 DKK
```

OUTPUT TERMINAL DEBUG CONSOLE

```
• ~/Dev/github.c/affrae/github-copilot-examples main !3 > cd python  
  08:21:07  
• ~/Dev/github.c/affrae/github-copilot-examples/python main !3 > python3  
  parse_expenses.py  
Traceback (most recent call last):  
  File "/Users/dfigucio/Developer/github.com/affrae/github-copilot-examples/py  
thon/parse_expenses.py", line 28, in <module>  
    expenses = parse_expenses (expenses_data)  
  File "/Users/dfigucio/Developer/github.com/affrae/github-copilot-examples/py  
thon/parse_expenses.py", line 18, in parse_expenses  
    date, value, currency = line.split(",")  
ValueError: not enough values to unpack (expected 3, got 1)  
• ~/Dev/github.c/affrae/github-copilot-examples/python main !3 > python3  
  parse_expenses.py  
2023-01-02 00:00:00 -34.01 USD  
2023-01-03 00:00:00 2.59 DKK  
2023-01-03 00:00:00 -2.72 EUR  
• ~/Dev/github.c/affrae/github-copilot-examples/python main !4 >
```

zsh python

zsh

Ln 15, Col 34 Spaces: 4 UTF-8 LF {} Python 3.9.6 64-bit CodeQuote - No Matches

P Performance

git commit -m "changes"



Writing

Useless Git Commit Messages

The screenshot shows a GitHub repository named 'python-app' by 'octocloudlabs'. The repository is private and has 2 watches, 0 forks, and 0 stars. The commit history shows a commit by 'chrisreddington' titled 'Add GitHub Copilot extension to devcontainer' with 11 commits. The commit message is 'Add GitHub Copilot extension to devcontainer'. The commit history table lists the following files and their commit messages:

File	Commit Message	Commit Date
.devcontainer	Add GitHub Copilot extension to devcontainer	last week
src	Add utils file	5 months ago
LICENSE	Initial commit	5 months ago
README.md	Update README.md	2 weeks ago

The README.md file content is displayed below the commit history. It features a section titled 'customer' with a link icon. The text describes the purpose of the code and includes a note about the customer ID generation process.

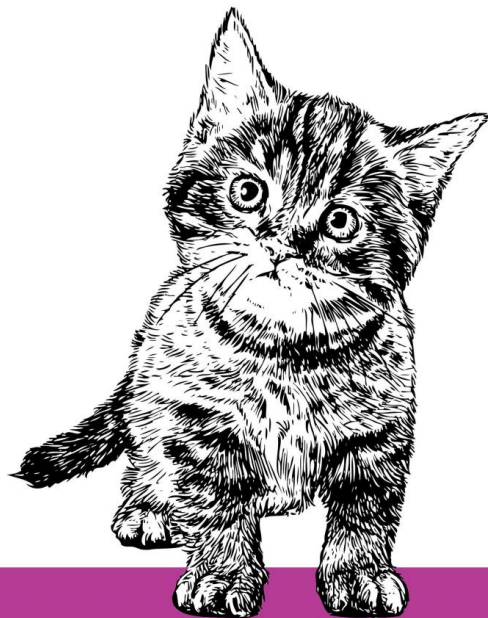
customer [🔗](#)

Example code to help demonstrate GitHub Copilot UI

Note: The way we generate customer ID's is an incrementing primary key on the `customer` table followed by a single digit checkdigit calculated using the Luhn algorithm which helps prevent simple customer id miskey errors.

The right sidebar contains the 'About' section, which states 'No description, website, or topics provided.' Below this are links for 'Readme', 'MIT license', 'Activity', '0 stars', '2 watching', and '0 forks'. The 'Releases' section states 'No releases published' and includes a link to 'Create a new release'. The 'Packages' section states 'No packages published' and includes a link to 'Publish your first package'. The 'Languages' section shows a bar chart with 'Python' at 100.0%.

How to actually learn any new programming concept



Essential

Changing Stuff and Seeing What Happens

Copilot Workspaces



E Efficiency and flow

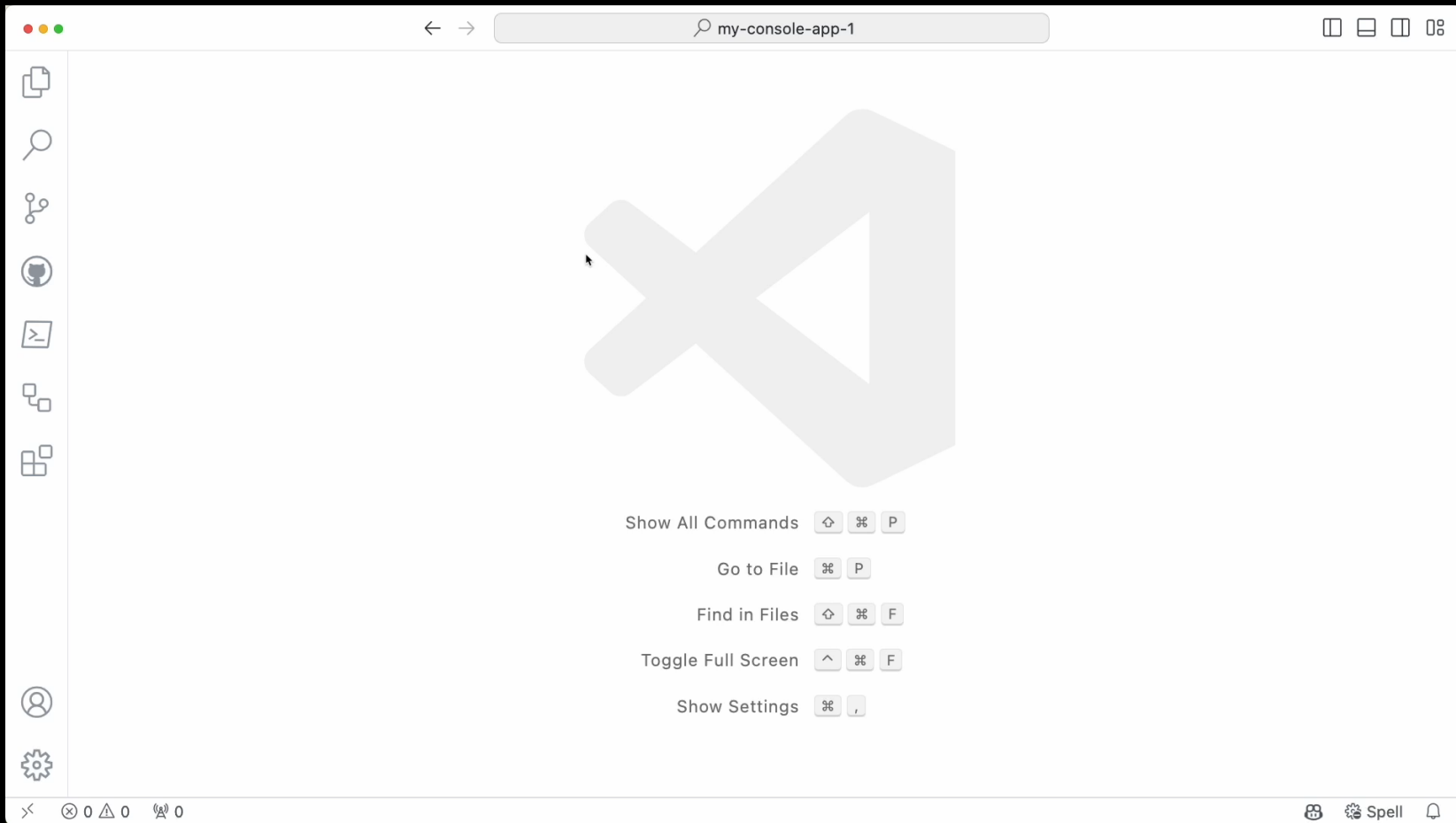
It would be a pure function if not for the side effects on your sanity



Turning Coffee Into Code

The Definitive Guide

Copilot Fix Issue



s Satisfaction and well-being

Combining slashes and dots until a thing happens



Expert

Regex by Trial and Error

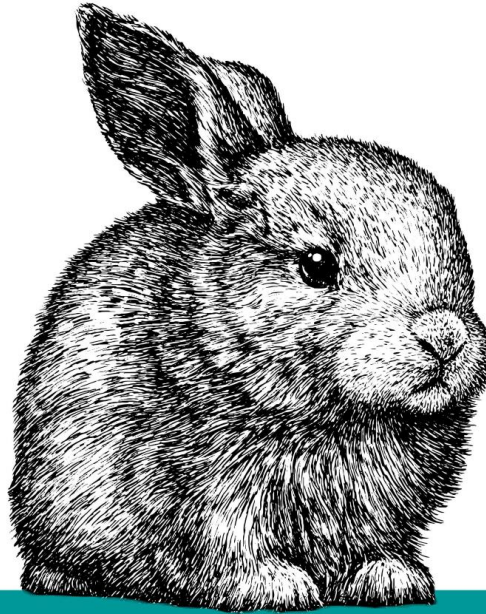
Copilot in CLI

```
chrisreddington@MBP ~ %
```

```
I
```

P Performance


Feigning knowledge of a word you've heard a few times



Expert

Pretending to Know About Stuff

chrisreddington



Chris Reddington
chrisreddington · he/him

Senior Manager, DevRel @ GitHub. The repos under my profile and @CloudWithChris are not endorsed in any way by my employer Demo Account: @chrisreddingt0n

Edit profile

102 followers · 104 following

GitHub

STAFF

United Kingdom

21:18 (UTC)

https://www.cloudwithchris.com

@reddobowen

chrisreddington / README.md

Projects

Chris Reddington

Hi! I'm Chris Reddington, Senior Manager for Enterprise Advocacy (International) at GitHub (aka I lead a team of advocates that help developers, engineering leads and execs at startups, enterprises and other organizations ❤️ GitHub). The repos under my profile and the CloudWithChris org are not endorsed in any way by my employer. Use at your own risk.

Followers 102

Follow @reddobowen

chrisreddington

Cloud with Chris

Cloud With Chris is my community-focused podcast, where we exploring Cloud concepts. We will regularly invite guests to talk about their experiences with the cloud and hear about some of the lessons learned around their cloud journey.

website Up

security headers

Please use an API Key from https://securityheaders.com/api

discussions 247 total

Discord 8 online

Subscribers 1.3k

Twitch offline

Hugo Creator Theme

Based on my work on Cloud With Chris, I decided to go a step further and contribute my site's theme to the OSS Community. Creator is a theme for the Hugo Static Site Generator, and is intended for content contributors. I absolutely welcome contributions, so if you're looking for a project to get into Open Source, or get your teeth into it Hugo - please take a look!

license MIT

last commit june

issues 2 open

About Me

Languages and Tools:

Communication and collaboration

Dev-first approach to productivity

github.co/34i4OIA



May 25, 2021 — Community, Company

Octoverse Spotlight 2021: The Good Day Project—Personal analytics to make your work days better

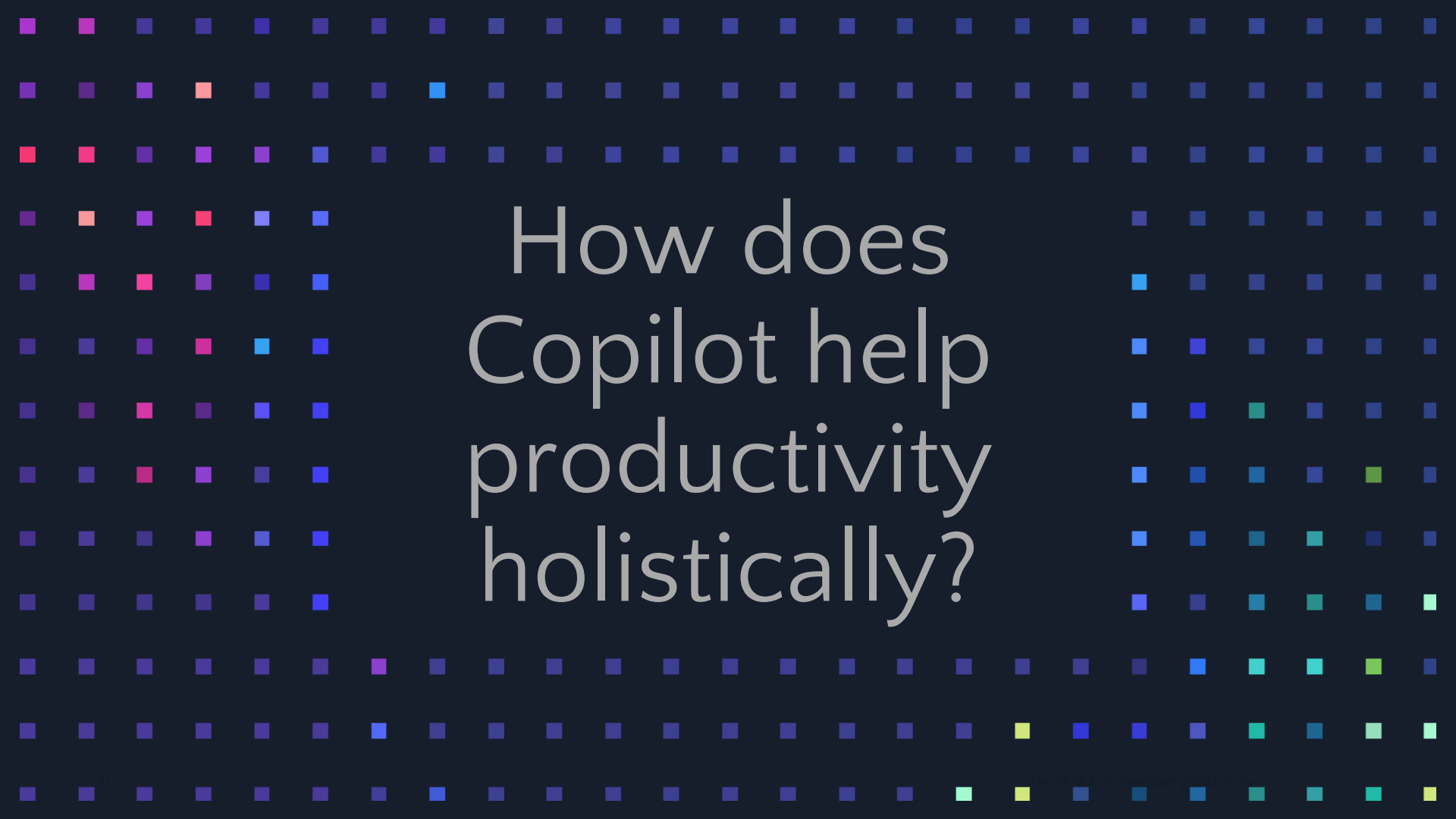


Eirini Kalliamvakou

“How is Copilot helping you?”

- 2.6K survey responses
- Users in the technical preview
- Vast majority professional programmers
- Assessed SPACE dimensions ++





How does
Copilot help
productivity
holistically?

Satisfaction & well-being

Performance

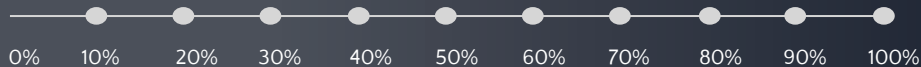


"Less frustrated when coding"

"More fulfilled with my job"

"Focus on more satisfying work"

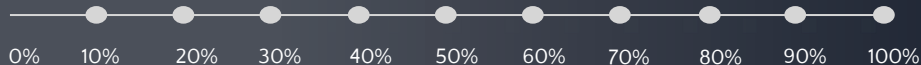
S



"Faster completion"

"Faster with repetitive tasks"

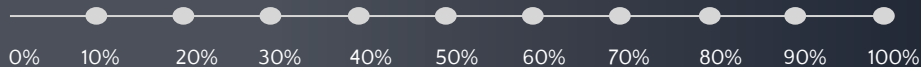
P



Efficiency & Flow



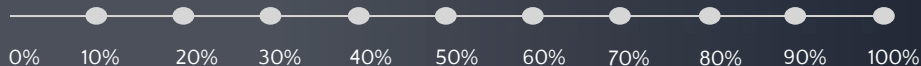
“Learn from suggestions”



"More in the flow"

“Less time searching”

“Less mental effort on repetitive tasks”



F

Benefits at all levels of the organization

*"The engineers' **satisfaction** with doing edgy things and us **giving them edgy tools** is a factor for me. Copilot makes things more exciting."*

– CTO

*"[With Copilot] I have to **think less**, and when I have to think it's the **fun stuff**. It sets off a little spark that makes coding more fun **and** more efficient. But we're not metric-driven, we're **emotion-driven**."*

– Eng. manager

*"It is almost like magic, it has more than once shocked me. Sometimes with a few characters it would give me the entire function. It's **built-in Stack Overflow** customized to you."*

– Senior Engineer

