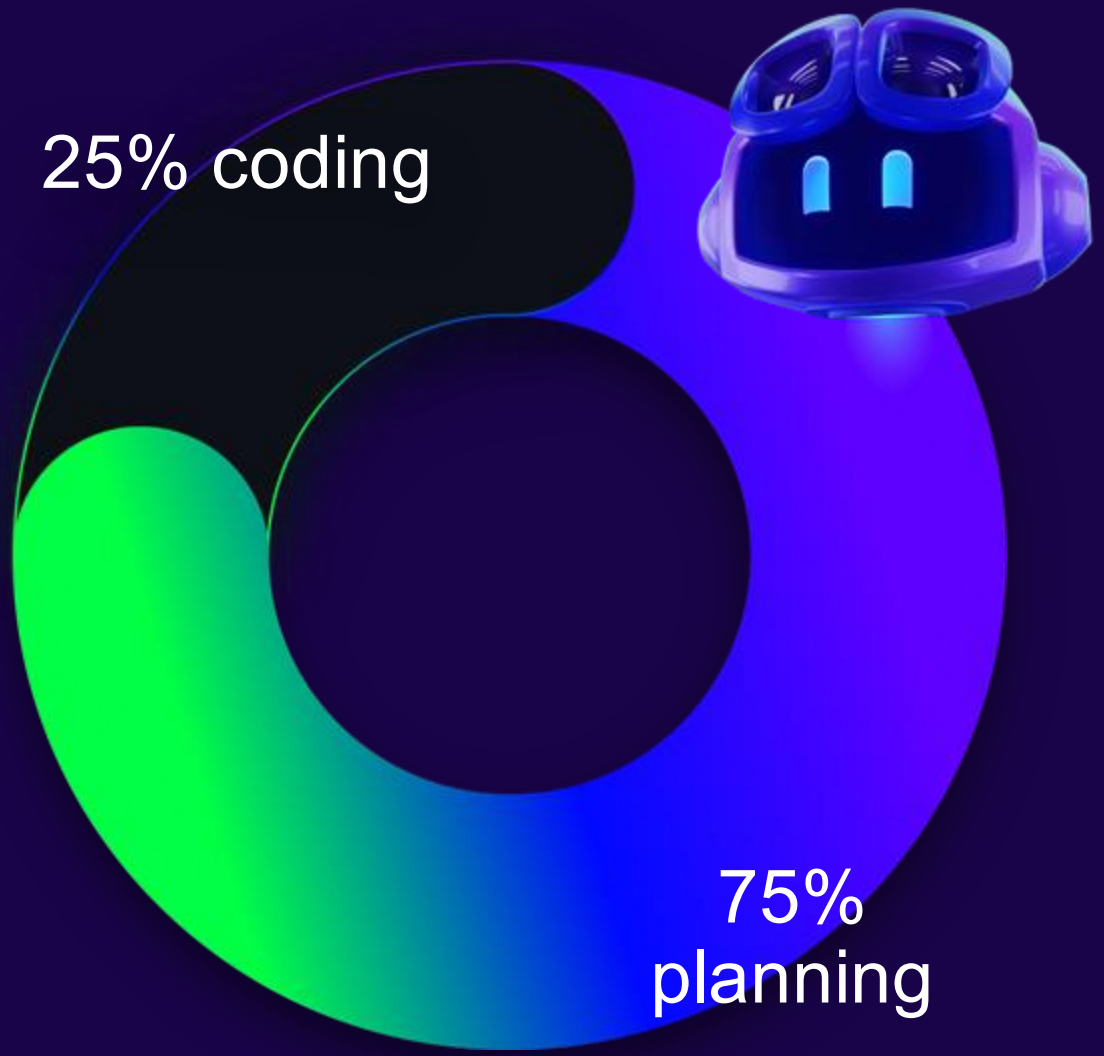
Beyond GitHub Copilot Tips and Tricks

April Yoho

Scubaninja

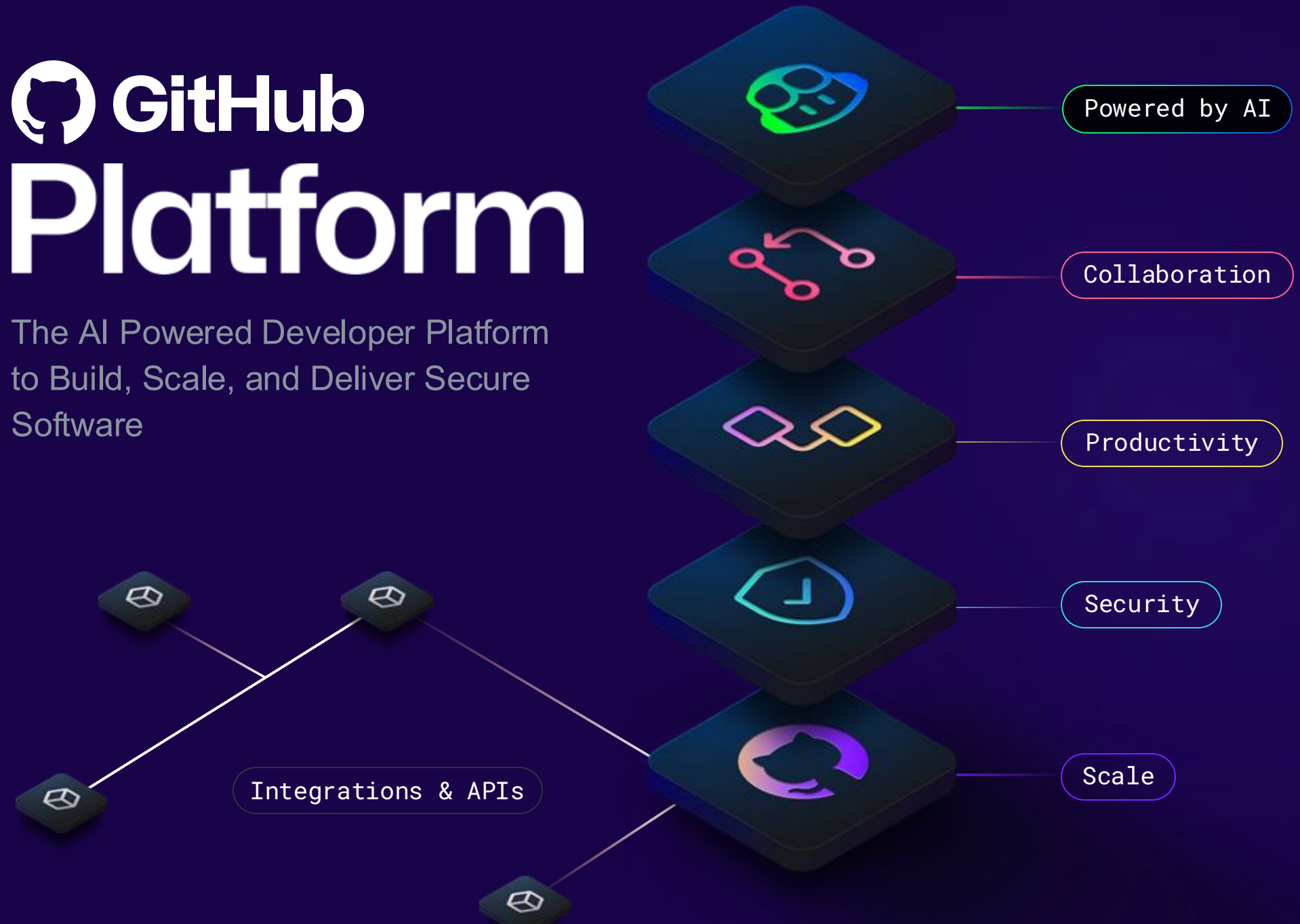
/azureapril

Developers dedicate 75% to non-coding tasks



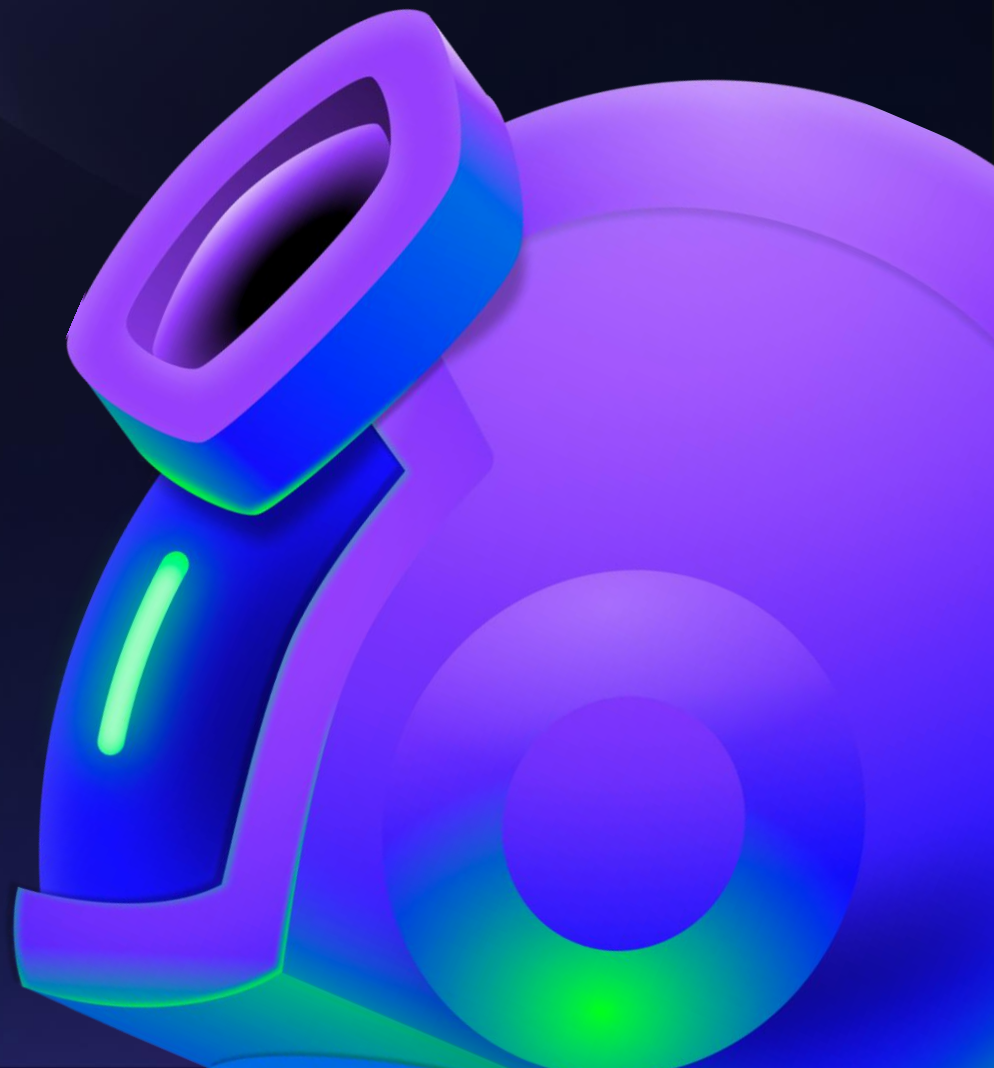
GitHub Platform

The AI Powered Developer Platform to Build, Scale, and Deliver Secure Software



GitHub

Hello!



We are building Copilot for the sake of developer happiness

GitHub Copilot

Built for developer happiness

55%

faster

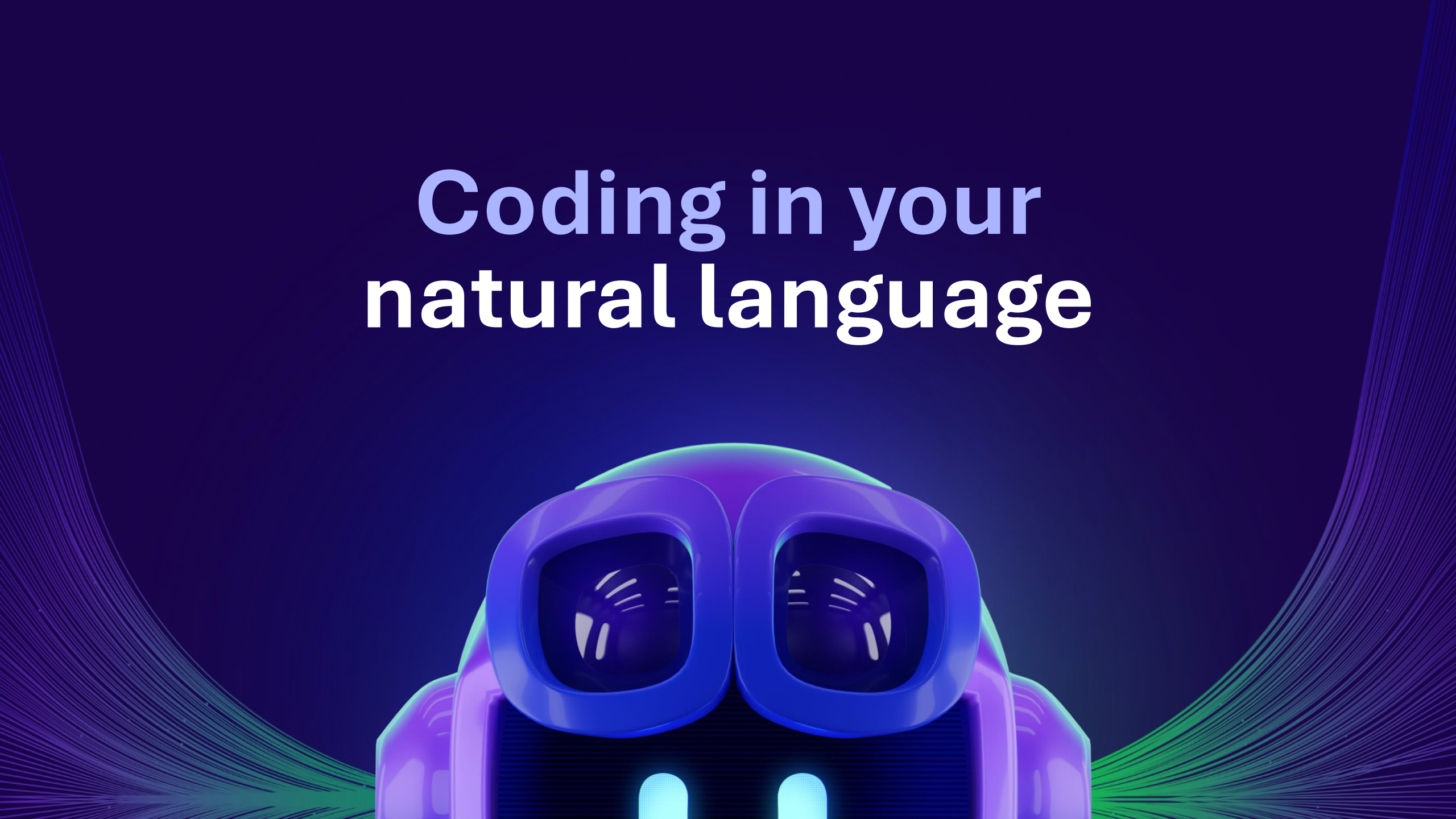
73%

were able to stay in the flow

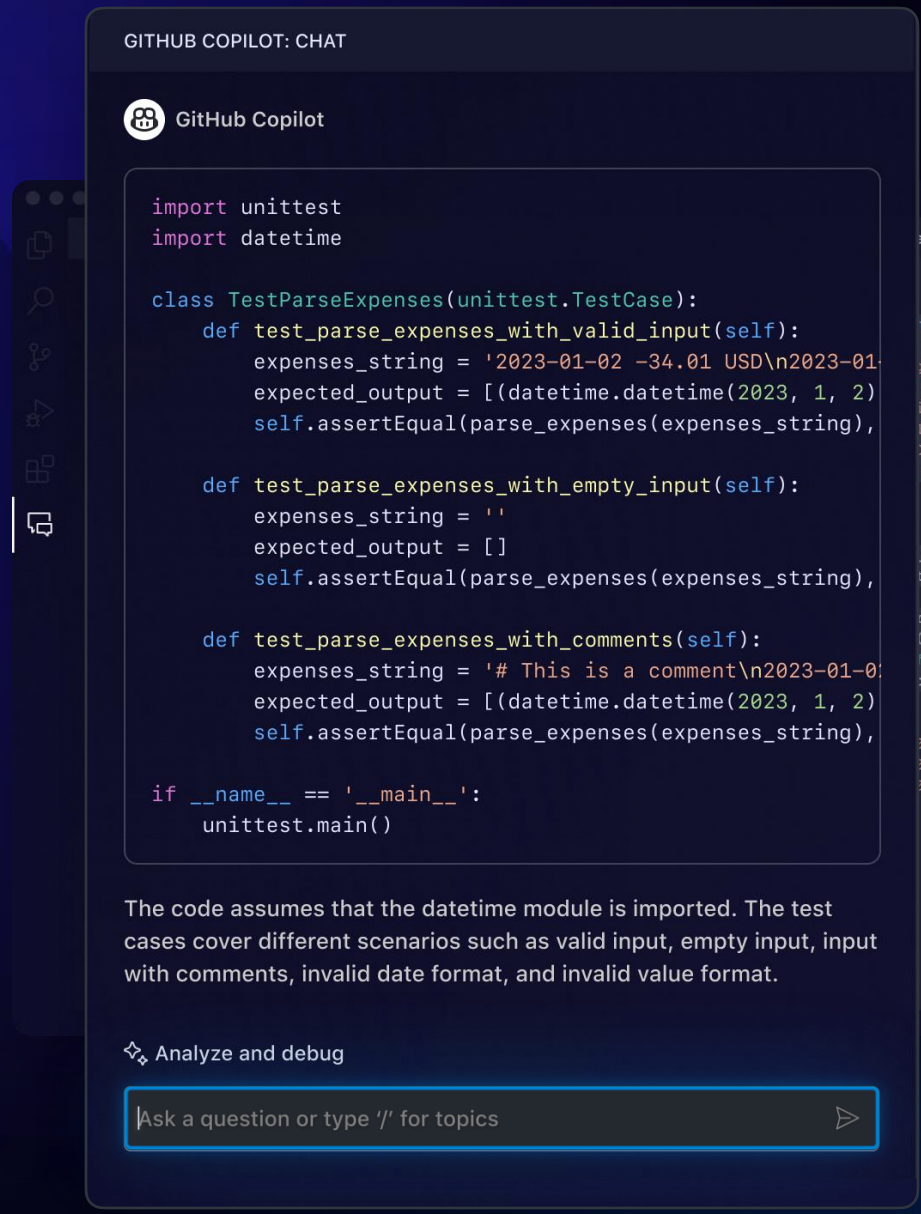
75%

felt more fulfilled in their jobs

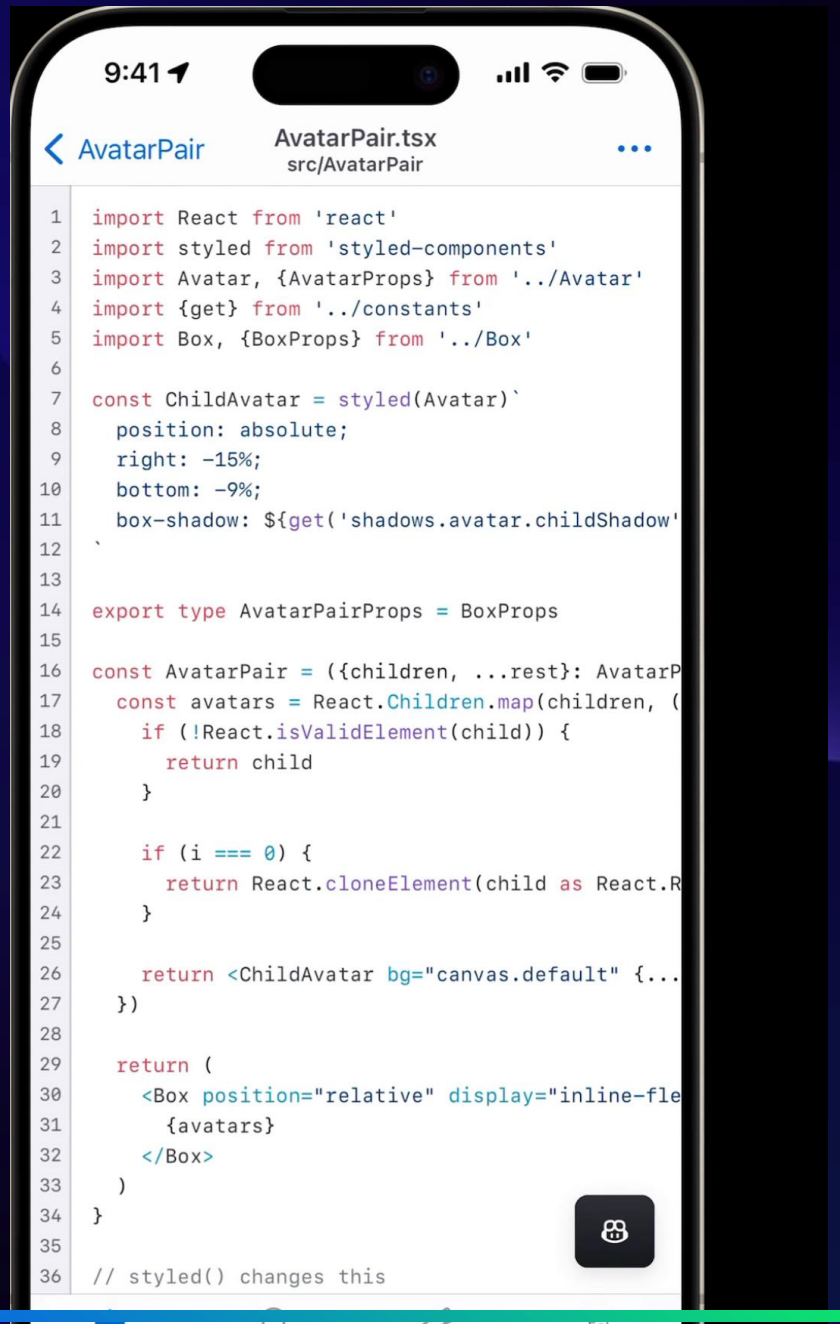
Coding in your natural language



Copilot Chat in VS & VS Code



Copilot Chat in Mobile



Demo

Recap

Limitations

Prompts and context

Trust but verify

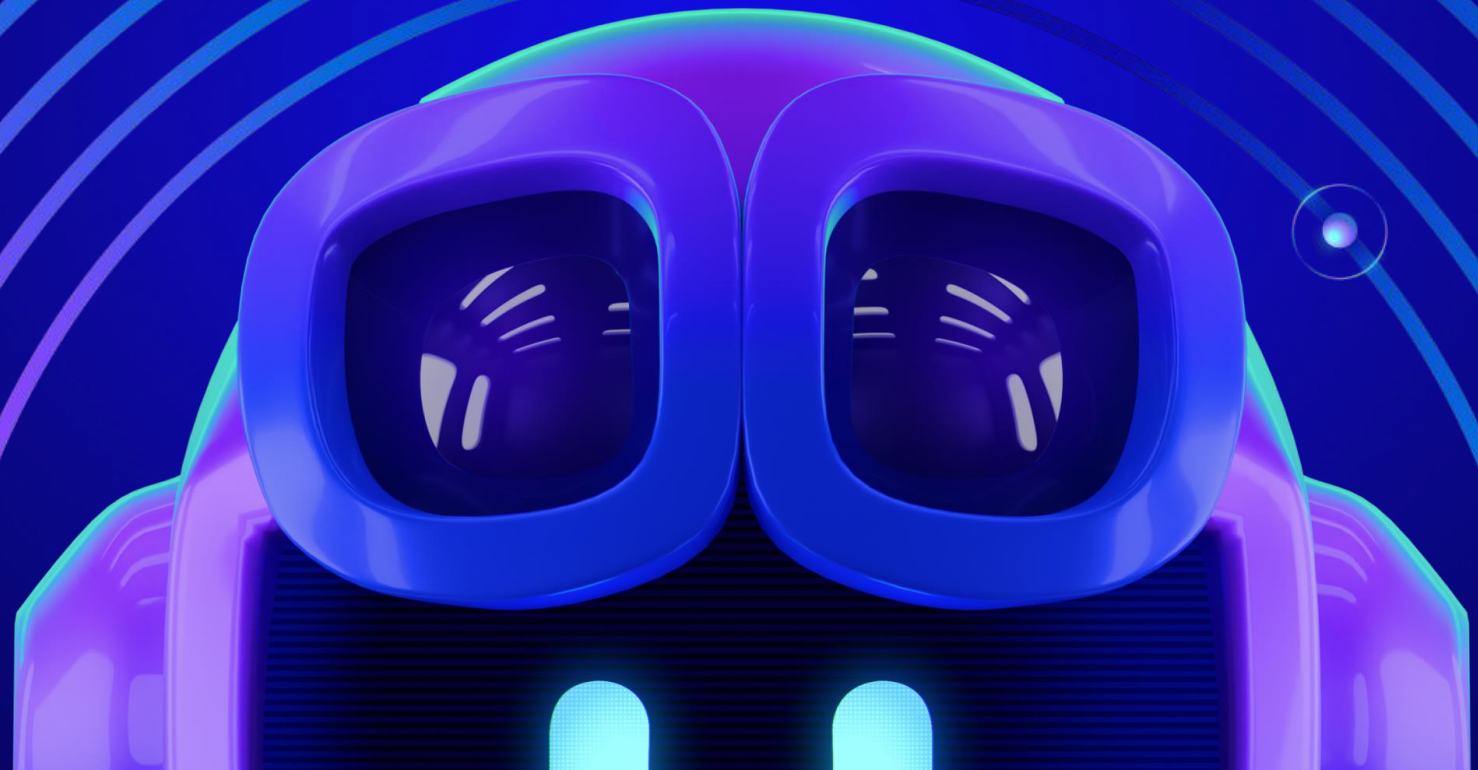

Technical Preview
Copilot Workspace
The Copilot–native dev environment, designed for everyday tasks


GitHub Copilot Workspace
Spec
Plan
Code
Build
Preview
Deploy
Issue
Repos
Codespaces
Actions
PR
Actions
GitHub Advanced Security

Demo: GitHub Copilot Workspace

PUBLIC PREVIEW

Copilot Extensions



Code

main

Jump to file

- models
  - __init__.py
  - book.py
  - user.py
- templates
- static
- database
- test
- app.py
- config.py
- requirements.txt
- README.md

bookstore_app / models / book.py

monalisa initial setup and book crud

bc07c5d · 23 days ago History

Code Blame 6339 lines (6339 sloc) · 1.05 MB Raw

```python
class Book:
    def __init__(self, isbn, title, author):
        self.isbn = isbn
        self.title = title
        self.author = author

    def __repr__(self):
        return f"<Book(isbn='{self.isbn}', title='{self.title}', author='{self.author}')>"

    @classmethod
    def find_books_by_author(cls, session, author_name):
        """
        List all books by a given author, ordered by title. Return a list of Book objects.
        """
        rows = session.execute('SELECT * FROM books WHERE author_name = %s ORDER BY title', [author_name])
        books = []
        for row in rows:
            books.append(cls(row.isbn, row.title, row.author_name))
        return books

    @classmethod
    def find_by_isbn(cls, session, isbn):
        """
        Retrieve a book by its ISBN. Return a Book object or None if not found.
        """
        row = session.execute('SELECT * FROM books WHERE isbn = %s', [isbn]).one()
        if row:
            return cls(row.isbn, row.title, row.author_name)
```

Recap

Build your internal community

Share tips, tricks and surprising moments

Iterate, iterate, iterate

Compliance & Security Questions

GitHub Copilot Trust Center

https://gh.io/ctc

Resources

GitHub Copilot Trust Center

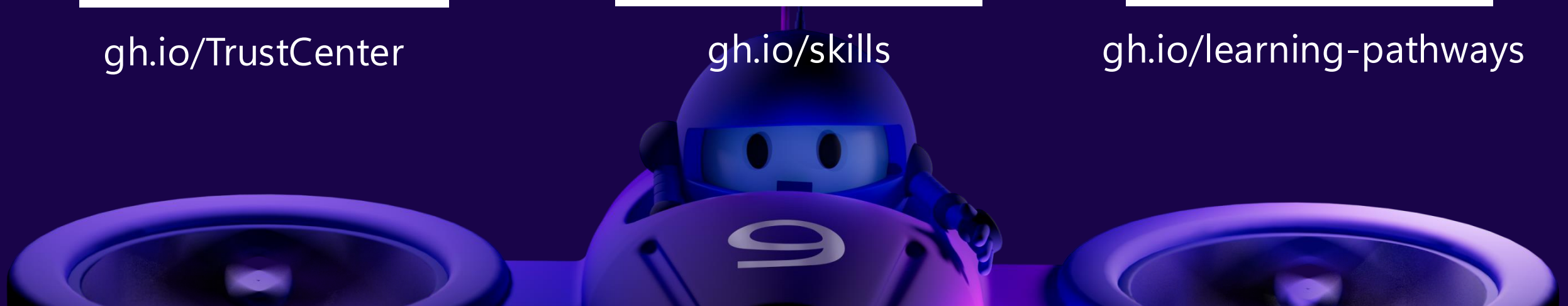
gh.io/TrustCenter

GitHub Interactive Courses

gh.io/skills

GitHub Learning Pathways

gh.io/learning-pathways



Thank you