

Collaborating with Agents in your Software Dev Workflow - Jon Peck & Christopher Harrison, Microsoft

 AI Engineer
140K subscribers

Subscribe

Like 2 | Dislike | Share | Clip | Save | ...

79 views Jun 27, 2025 AIEWF 2025 Complete Playlist

GitHub Copilot's agentic capabilities enhance its ability to act as a peer programmer. From the IDE to the repository, Copilot can generate code, run tests, and perform tasks like creating pull requests using Model Context Protocol (MCP). This instructor-led lab will guide you through using agent capabilities on both the client and the server. Key takeaways include:

Understanding how to bring agents into your software development workflow

Identifying scenarios where agents can be most impactful, as well as tips and tricks to provide the right context to lead to success

Discovering how Model Context Protocol provides access to an additional set of external tools and capabilities that the agent can use

Recommended practices to accelerate your development while maintaining code quality.

About Jon Peck

An Enterprise Advocate (and occasional manager) at GitHub, Jon Peck meets daily with maintainers, startups, and F500 executives to familiarize them with industry best practices, policy suggestions, and product capabilities across DevOps and AI. With 25+ years of experience as a fullstack developer, architect, and advocate, he aims to to bring engaging, real-world learnings to both boardrooms and global conferences.

- Speaker (conferences): Dev Exec World 2025, STARWEST 2024, InnerSource Summit 2023, GitHub Galaxy 2023, DevWeek Management 2023, Startup Grind 2022, GitHub InFocus 2022, DeveloperWeek 2018-20, SeattleJS, Global AI Conf 2018-19, AI Next 2019-20, MLOps World, Data Innovation Summit, Nordic APIs 2018-19 (keynote), ODSC East+West, API World, O'Reilly AI, OSCON
- Speaker (tech schools): Galvanize, CodeFellows, Metis, Epicodus, Alchemy
- Organizer: Seattle Building Intelligent Applications Meetup
- Educator: Cascadia College, Seattle C&W, consultant
- Lead Developer: Empower Engine, Giftstarter, Mass General Hospital, Cornell University
- Technical Advocate: Algorithmia, GitHub

About Christopher Harrison

Christopher is a long-time geek who's spent the bulk of his career training, supporting and upskilling developers. He's a web developer at heart with passions which span from Python to DevOps to TypeScript to AI. In his current role as an Enterprise Advocate for GitHub he seeks to help organizations improve their DevOps process and culture. When not found writing code he can be found running, playing Civilization, or spending time with his partner and their four-legged child (a rescue mutt).

Recorded at the AI Engineer World's Fair in San Francisco. Stay up to date on our upcoming events and content by joining our newsletter here:

<https://www.ai.engineer/newsletter>

Agenda

- 01 Context
- 02 Model Context Protocol
- 03 Providing context with Instruction files
- 04 Copilot in the IDE
- 05 Copilot coding agent
- 06 Recap



Context

What is Context?

```
// FormatYearRange returns a string representing a year range.
func FormatYearRange(startYear, endYear int) string {
    if startYear == endYear {
        return fmt.Sprintf("%d", startYear)
    }
    // Use YYYY-YY format
    return fmt.Sprintf("%d-%d", startYear, endYear)
}
```

```
EmptyBlock = ' ' // Represents empty space
FutureBlock = '.' // Represents a future state

// Foundation blocks (bottom)
FoundationLow = '0' // 1-9
FoundationMed = '0' // 34-67
FoundationHigh = '0' // 67-99

// Middle blocks (intermediate)
MiddleLow = '0'
```

OPEN EDITORS	cmd
GH-SKYLINE	1
.devcontainer	2
.github	3
.vscode	4
cmd	5
skyline	6
skyline_test.go	7
skyline.go	8
root_test.go	9
root.go	10
internal	11
.gitignore	12
.golangci.yaml	13
	14
	15
	16

Readable Code

GitHub Copilot understands natural language. Code is its language.

Comments

As with fellow developers, comments explain how and why things are done a certain way.

Project Structure

Separation of concerns, structure and modularity help bring clarity to our thought process.

Context

Help Copilot help you

The screenshot shows four panels side-by-side:

- Context:** A code editor window showing a Go file with imports like `#editor`, `#file`, `#selection`, etc.
- Intent:** A text input field asking for an explanation of a Python developer's Go code.
- Clarity:** A text input field asking to write an HTML parser with specific requirements.
- Specificity:** A text input field asking to create a reusable method with specific constraints.

Context	Intent	Clarity	Specificity
Information provided to help GitHub Copilot better understand the task	The specific goal or purpose you have in mind when creating a prompt	The quality of being easy to understand	The level of detail and precision in your prompt

The screenshot shows three main sections:

- Scope Of Changes:** Includes icons for a search icon, a clock icon, a circular arrow icon, and a code editor icon.
- Completions:** Shows timeframes: "Next Few Lines", "Hundreds of Milliseconds", "Inner Loop", and "VS Code (Editor)".
- Chat/Edits:** Shows timeframes: "Seconds", "Inner Loop", "VS Code (Chat)", and "Multi-File Edits".
- Agent Mode:** Shows timeframes: "Complete tasks", "Minutes", "Inner Loop", and "VS Code (Chat)".
- Copilot coding agent:** Shows scope levels: "Entire issues", "Tens of minutes", "Outer Loop", and "GitHub.com".

The screenshot shows three main sections:

- Give Copilot a task:** A card with fields for Assignees (Copilot), Labels (enhancement, mobile, ui), and Projects (Copilot Airways Backlog). Status is set to In Progress.
- Track the agent's progress:** A card showing a terminal session output and a summary: "Analysis of Current Components and Needed Improvements" for "Titlebar.tsx".
- Review the agent's work:** A card titled "Changes Made" listing improvements for "Header (titlebar.tsx) Improvements" and "Footer (footer.tsx) Improvements".

Getting the most out of /

 Copilot coding agent

Description

The current header (TitleArea.jsx) and footer (Footer.jsx) components need improvements to fit responsiveness on mobile devices. Based on review of the codebase, these components need adjust spacing, and possibly different UI organization for smaller screens to enhance the mobile user experience.

Why this is important

- Improves user experience across different device sizes
- Ensures consistent branding and navigation regardless of device
- Helps maintain professional appearance on all screen sizes
- Increases accessibility for users on mobile devices

Acceptance Criteria

- Review and update ./components/ui/TitleArea.jsx - to use responsive design patterns
- Implement a mobile-friendly navigation menu (potentially a hamburger menu for small screens)
- Ensure proper spacing and alignment of navigation items on mobile
- Ensure display of active page indicators on mobile view
- Review and update ./components/ui/Footer.jsx - for better mobile experience:
- Update footer column layout to stack on mobile devices
- Adjust padding and margins for better readability on small screens
- Ensure all interactive elements meet touch target size requirements (min 40pxx40px)

This is a Next.js-based travel application with TypeScript that helps users search for trips. The application uses React components, server components, and client components as these guidelines when contributing.

Code Standards

Required Before Each Commit

- Run `npm run lint` to ensure code follows project standards
- Make sure all components follow Next.js App Router patterns
- Client components should be marked with 'use client' when they use browser APIs
- Server components (default in Next.js App Router) should be used where possible
- When adding new functionality, make sure you update the README
- Make sure that the repository structure documentation is correct and accurate in the README
- Ensure all tests pass by running `npm run test` in the terminal

TypeScript and React Patterns

- Use TypeScript interfaces/types for all props and data structures
- Follow React best practices (hooks, functional components)
- Use proper state management techniques
- Components should be modular and follow single-responsibility principle

name: "Copilot Setup Steps"

```
# Allows you to test the setup steps from your repository
on: workflow_dispatch

jobs:
  copilot-setup-steps:
    runs-on: ubuntu-latest
    # Set the permissions to the lowest permissions possible
    # If you want to clone the repository as part of your
    # workflow, add the following to your workflow file
    # contents: read
    steps:
      - name: Checkout code
        uses: actions/checkout@11bd71901bbe5b1e0ceea7

      - name: Set up Node.js
        uses: actions/setup-node@49933ea5288caeac8642c
        with:
          node-version: '22'
```

Clear and well-scoped issues

Provide context with Custom instructions

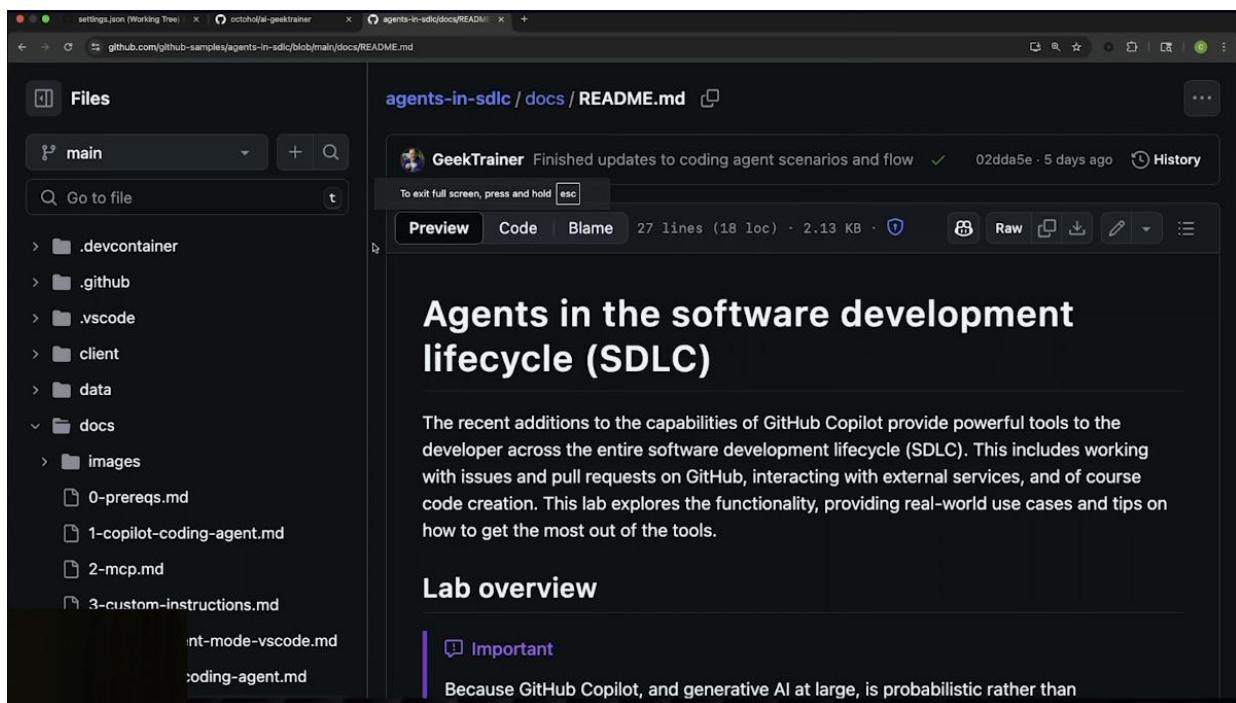
Customize the dev environment

2. Navigate to <https://github.com/octohol>

3. Select "accept invitation"

Get started with the lab

<https://gh.io/agent-sdlc-lab>



agents-in-sdlc / docs / README.md

GeekTrainer Finished updates to coding agent scenarios and flow 02ddae · 5 days ago History

To exit full screen, press and hold esc

Preview Code Blame 27 lines (18 loc) · 2.13 KB

Agents in the software development lifecycle (SDLC)

The recent additions to the capabilities of GitHub Copilot provide powerful tools to the developer across the entire software development lifecycle (SDLC). This includes working with issues and pull requests on GitHub, interacting with external services, and of course code creation. This lab explores the functionality, providing real-world use cases and tips on how to get the most out of the tools.

Lab overview

Important Because GitHub Copilot, and generative AI at large, is probabilistic rather than

The screenshot shows a GitHub repository page for 'ai-geektrainer'. The 'Code' tab is selected. The commit history lists several initial commits from 'GeekTrainer' across various files and folders: '.devcontainer', '.github', '.vscode', 'client', and 'README'. All commits were made 5 hours ago. The repository has 3 branches and 0 tags. The 'About' section indicates no description, website, or topics provided.

The screenshot shows a GitHub pull request page for 'ai-geektrainer' with the title 'Add CRUD endpoints for games API with comprehensive error handling and tests #3'. The pull request is marked as 'Draft'. The 'New Endpoints Added' section contains two API endpoint descriptions: 'POST /api/games' and 'PUT /api/games/'. The 'Assignees' sidebar shows suggestions for assigning the pull request, with 'Copilot Your AI pair programmer' selected.

New Endpoints Added

POST /api/games

Creates a new game with validation for all required fields:

```
{  
    "title": "New Game Title",  
    "description": "Game description (min 10 chars)",  
    "publisher_id": 1,  
    "category_id": 1,  
    "star_rating": 4.5  
}
```

PUT /api/games/

↳ an existing game with partial update support:

Assignees

Assign up to 10 people to this pull request

Type or choose a user

Clear assignees

✓ Copilot Your AI pair programmer

✓ GeekTrainer Christopher Harrison

Suggestions

chrisreddington Chris Reddington (busy)

Damovisa Damian Brady

joshjohanning Josh Johanning (busy)

A screenshot of a GitHub issue page. The issue is titled "Create endpoints for creating and editing games #2". The description reads: "We're going to be creating functionality in the future to allow for the submission (and editing) of games. For now we just want the endpoints so we can explore how we want to create the UX and do some acceptance testing. Our requirements are:" followed by a bulleted list of requirements. A note at the bottom says "Copilot linked a pull request that will close this issue 4 hours ago" with a link to "#3". On the right side, there are settings for assignees (Copilot), labels (No labels), type (No type), projects (No projects), milestone (No milestone), and relationships (None yet). At the bottom left, there's a "Create sub-issue" button.

A screenshot of a GitHub issue page, identical to the one above, showing the same issue details and Copilot assignment. The requirements list is present, and the note about the pull request closing the issue is also present.

A screenshot of a GitHub repository page for "octohol / ai-geektrainer". The repository has 1 issue and 2 pull requests. The "Issues" tab is selected. An "Open" button is visible next to the issue count. Below the tabs, there's a "New issue" button. The main content area shows the issue "Create endpoints for creating and editing games #2" with its title, description, and an "Open" button.

A screenshot of a GitHub repository page for "octohol / ai-geektrainer". The repository has 0 stars, 0 forks, 0 watching, 3 branches, 0 tags, and activity. It includes links to "MIT license", "Code of conduct", and "Security policy". There are also buttons for "Custom properties" and "Edit".

GeekTrainer	Initial commit	140409e · 6 hours ago
.devcontainer	Initial commit	6 hours ago
.github	Initial commit	6 hours ago
.vscode	Initial commit	6 hours ago
client	Initial commit	6 hours ago
data	Initial commit	6 hours ago

GeekTrainer	Initial commit	140409e · 6 hours ago
Name	Last commit message	Last commit date
..		
instructions	Initial commit	6 hours ago
workflows	Initial commit	6 hours ago
CODEOWNERS	Initial commit	6 hours ago
ot-instructions.md	Initial commit	6 hours ago

GeekTrainer	Initial commit	140409e · 6 hours ago
Name	Last commit message	Last commit date
..		
copilot-setup-steps.yml	Initial commit	6 hours ago
run-tests.yml	Initial commit	6 hours ago

ai-geektrainer / .github / workflows / copilot-setup-steps.yml

GeekTrainer Initial commit 140409e · 6 hours ago

37 lines (31 loc) · 1.26 KB

Code **Blame**

```
name: "Copilot Setup Steps"

# Allows you to test the setup steps from your repository's "Actions"
on: workflow_dispatch
```

A screenshot of a GitHub Copilot code editor window. The tab bar at the top shows 'Code' and 'Blame'. The main area displays a YAML file named 'copilot-setup-steps.yml' with the following content:

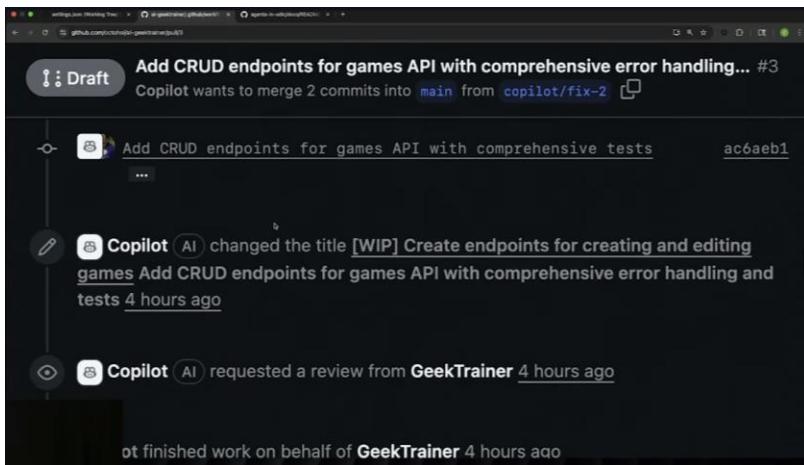
```
name: "Copilot Setup Steps"
on: workflow_dispatch
jobs:
  copilot-setup-steps:
    runs-on: ubuntu-latest
    # Set the permissions to the lowest permissions possible needed for
    permissions:
      # If you want to clone the repository as part of your setup step
      contents: read
    steps:
      - name: Checkout code
```

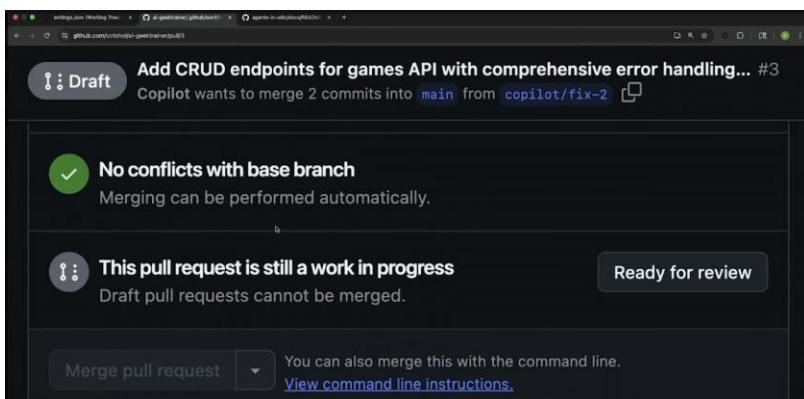
A screenshot of a GitHub repository page for 'octohol/ai-geektrainer'. The navigation bar includes 'Code', 'Issues (1)', 'Pull requests (2)', 'Actions', 'Projects', and a dropdown for branches ('main'). The main content shows a commit by 'GeekTrainer' titled 'Initial commit' made 6 hours ago. The commit message is empty. It contains 37 lines (31 loc) and is 1.26 KB in size.

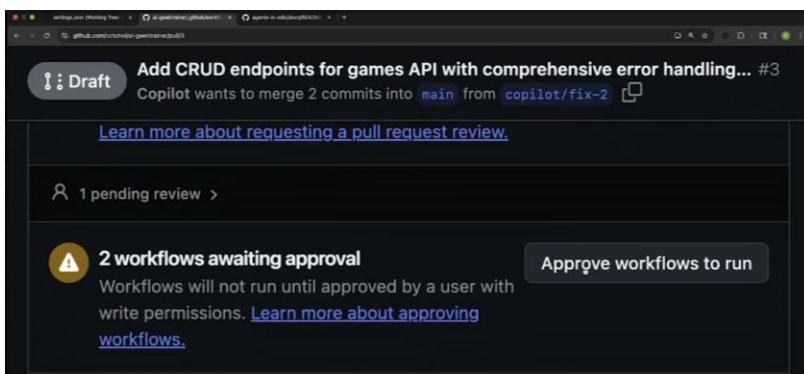
A screenshot of a GitHub pull request list. The header shows '2 Open' and '0 Closed'. The list contains two items:

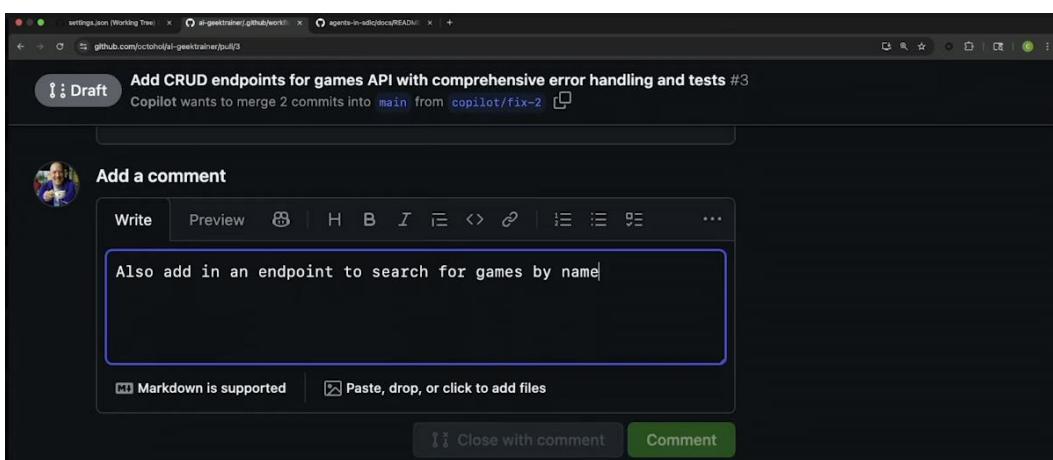
- Add CRUD endpoints for games API with comprehensive error handling and tests** (opened 4 hours ago by Copilot AI)
- Bump svelte from 5.33.12 to 5.33.14 in /client in the npm-production group** (opened 6 hours ago by dependabot bot)

A screenshot of a GitHub pull request page for 'octohol/ai-geektrainer/pull/3'. The navigation bar includes 'Code', 'Issues (1)', 'Pull requests (2)', 'Actions', 'Projects', and a dropdown for branches ('main'). A message at the top says 'Copilot requested your review on this pull request.' with a 'Add your review' button. The pull request details show a title 'Add CRUD endpoints for games API with comprehensive error handling and tests #3' and a note 'Copilot wants to merge 2 commits into main from copilot/fix-2'. The bottom of the page has 'Edit', 'Jump to bottom', and a profile picture.

A screenshot of a GitHub pull request page. The title is "Add CRUD endpoints for games API with comprehensive error handling... #3". It's marked as a draft. A message says "Copilot wants to merge 2 commits into main from copilot/fix-2". Below this, a commit titled "Add CRUD endpoints for games API with comprehensive tests" is shown, with a link to "ac6aeb1". A note indicates "Copilot AI changed the title [WIP] Create endpoints for creating and editing games Add CRUD endpoints for games API with comprehensive error handling and tests 4 hours ago". Another note says "Copilot AI requested a review from GeekTrainer 4 hours ago". At the bottom, it says "Copilot finished work on behalf of GeekTrainer 4 hours ago".

A screenshot of the same GitHub pull request page. It now shows a green checkmark icon and the text "No conflicts with base branch". It says "Merging can be performed automatically." Below this, a note says "This pull request is still a work in progress" and "Draft pull requests cannot be merged." There is a "Ready for review" button. At the bottom, there are buttons for "Merge pull request" and "View command line instructions".

A screenshot of the GitHub pull request page. It shows a note "Learn more about requesting a pull request review." Below this, it says "1 pending review >". A yellow warning icon indicates "2 workflows awaiting approval". It says "Workflows will not run until approved by a user with write permissions." and provides a link "Learn more about approving workflows.". There is a "Approve workflows to run" button.

A screenshot of the GitHub pull request page. It shows a "Add a comment" section. A user profile picture is visible. The text input field contains "Also add in an endpoint to search for games by name". Below the input field, it says "Markdown is supported" and "Paste, drop, or click to add files". At the bottom, there are "Close with comment" and "Comment" buttons.

[settings.json \(Working Tree\)](#) [ai-geektrainer/github/work!](#) [agents-in-edc/docs/README](#) | +

Add CRUD endpoints for games API with comprehensive error handling and tests #3
Copilot wants to merge 2 commits into `main` from `copilot/fix-2`

Copilot finished work on behalf of **GeekTrainer** 4 hours ago

 **GeekTrainer** commented now

Also add in an endpoint to search for games by name

1 reply

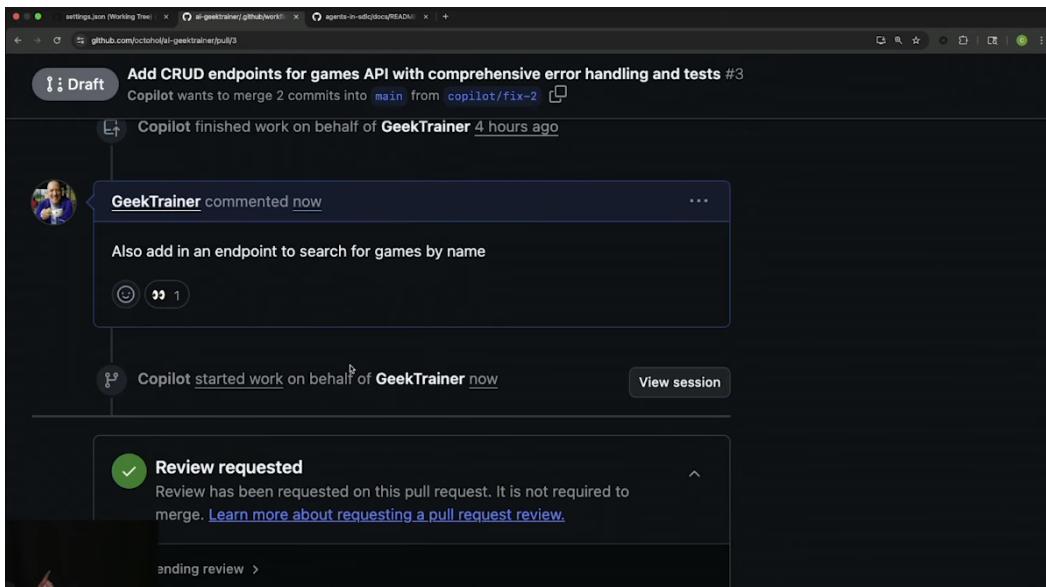
 1 reply

Copilot started work on behalf of **GeekTrainer** now

[View session](#)

 **Review requested**
Review has been requested on this pull request. It is not required to merge. [Learn more about requesting a pull request review.](#)

[Pending review >](#)



[settings.json \(Working Tree\)](#) [Sessions - Add CRUD endpoints](#) [agents-in-edc/docs/README](#) | +

[github.com/octoholai-geektrainer/pull/3/agent-sessions/ed6053ae-70f7-417b-b25c-beff39f688e7](#)

 octoholai / **ai-geektrainer**

[Code](#) [Issues 1](#) [Pull requests 2](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [...](#)

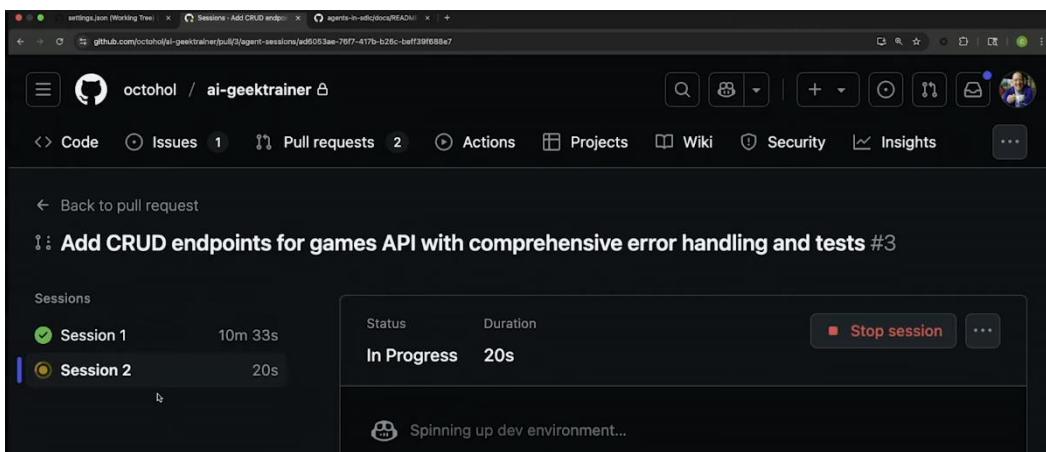
[← Back to pull request](#)

Add CRUD endpoints for games API with comprehensive error handling and tests #3

Sessions

	Status	Duration	Actions
 Session 1	In Progress	10m 33s	Stop session ...
 Session 2	In Progress	20s	Stop session ...

 Spinning up dev environment...



[settings.json \(Working Tree\)](#) [Sessions - Add CRUD endpoints](#) [agents-in-edc/docs/README](#) | +

[github.com/octoholai-geektrainer/pull/3/agent-sessions/ed6053ae-70f7-417b-b25c-beff39f688e7](#)

 octoholai / **ai-geektrainer**

[Code](#) [Issues 1](#) [Pull requests 2](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [...](#)

[← Back to pull request](#)

Add CRUD endpoints for games API with comprehensive error handling and tests #3

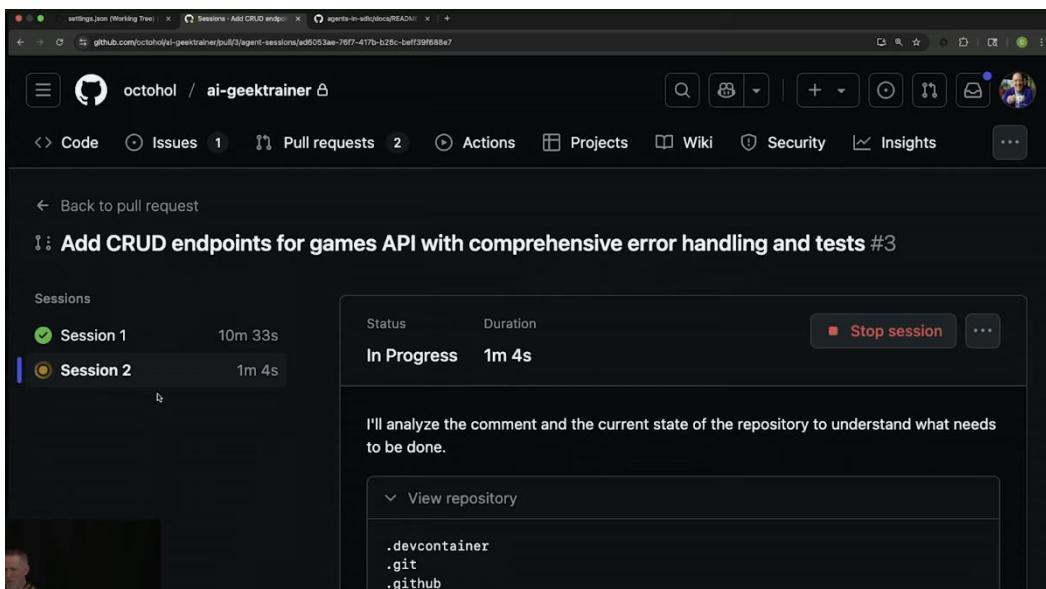
Sessions

	Status	Duration	Actions
 Session 1	In Progress	10m 33s	Stop session ...
 Session 2	In Progress	1m 4s	Stop session ...

I'll analyze the comment and the current state of the repository to understand what needs to be done.

[View repository](#)

```
.devcontainer  
.git  
.github
```



Draft Add CRUD endpoints for games API with comprehensive error handling and tests #3
Copilot wants to merge 2 commits into `main` from `copilot/fix-2`

Copilot finished work on behalf of **GeekTrainer** 4 hours ago

GeekTrainer commented 4 minutes ago
Also add in an endpoint to search for games by name

Copilot started work on behalf of **GeekTrainer** 4 minutes ago [View session](#)

Review requested
Review has been requested on this pull request. It is not required to merge. [Learn more about requesting a pull request review.](#)

EXPLORER

- AI-GEEKTRAINER [CODES...]
- .devcontainer
- .github
 - instructions
 - flask-endpoint.instructions.md
 - python-tests.instructions.md
 - ui.instructions.md
 - workflows
 - CODEOWNERS
 - copilot-instructions.md
 - dependabot.yml
- .vscode
- mcp.json
- settings.json
- client
- data
- docs
- scripts
- server

```
flask-endpoint.instructions.md    {} settings.json (Working Tree) x app.py ...  
.vscode > {} settings.json > ...  
11   "cSpell.words": [  
12     "isouter",  
13     "prebuild",  
14     "SDLC"  
15   ],  
16   "github.copilot.nextEditSuggestions.enabled": true,  
17   "editor.tabSize": 4,  
18   "chat.agent.enabled": true  
19 }
```

EXPLORER

- AI-GEEKTRAINER [CODES...]
- .devcontainer
- .github
 - instructions
 - flask-endpoint.instructions.md
 - python-tests.instructions.md
 - ui.instructions.md
 - workflows
 - CODEOWNERS
 - copilot-instructions.md
 - dependabot.yml
- .vscode
- mcp.json
- settings.json
- client
- data
- docs
- scripts
- server

```
flask-endpoint.instructions.md    copilot-instructions.md x app.py games.py ...  
.github > copilot-instructions.md  
1 # Tailspin Toys Crowd Funding Development Guidelines  
2  
3 This is a crowdfunding platform for games with a developer theme. The app  
4  
5 ## Code standards  
6  
7 ### Required Before Each Commit  
8  
9 - Run Python tests to ensure backend functionality  
10 - For frontend changes, run builds in the client directory to verify build  
11 - When making API changes, update and run the corresponding tests to ensure  
12 - When updating models, ensure database migrations are included if needed  
13 - When adding new functionality, make sure you update the README  
14 - Make sure all guidance in the Copilot Instructions file is updated with  
15  
16 ## Global language guidance  
17  
18 - Use type hints for function parameters and return values for all languages  
19  
20 ## Python and Flask Patter... Extension activation failed, run the 'Developer: Toggle Developer Tools' command  
21  
22 - Use SQLAlchemy models for database interactions
```

```
 1 # Tailspin Toys Crowd Funding Development Guidelines
 2
 3 This is a crowdfunding platform for games with a developer theme.
 4 The application uses a Flask backend API with SQLAlchemy ORM for
 5 database interactions, and an Astro/Svelte frontend with Tailwind
 6 CSS for styling. Please follow these guidelines when contributing:
 7
 8 ## Code standards
 9
10 ### Required Before Each Commit
11 - Run Python tests to ensure backend functionality
12 - For frontend changes, run builds in the client directory to
13 verify build success and the end-to-end tests, to ensure
14 everything works correctly
15 - When making API changes, update and run the corresponding tests
16 to ensure everything works correctly
17 - When updating models, ensure database migrations are included
18 if needed
19 - When adding new functionality, make sure you update the README
20 - Make sure all guidance in the Copilot Instructions file is
21 updated with any relevant changes, including to project structure
```

```
 7 ### Required Before Each Commit
 8
 9 - Run Python tests to ensure backend functionality
10 - For frontend changes, run builds in the client directory to
11 verify build success and the end-to-end tests, to ensure
12 everything works correctly
13 - When making API changes, update and run the corresponding tests
14 to ensure everything works correctly
15 - When updating models, ensure database migrations are included
16 if needed
17 - When adding new functionality, make sure you update the README
18 - Make sure all guidance in the Copilot Instructions file is
19 updated with any relevant changes, including to project structure
```

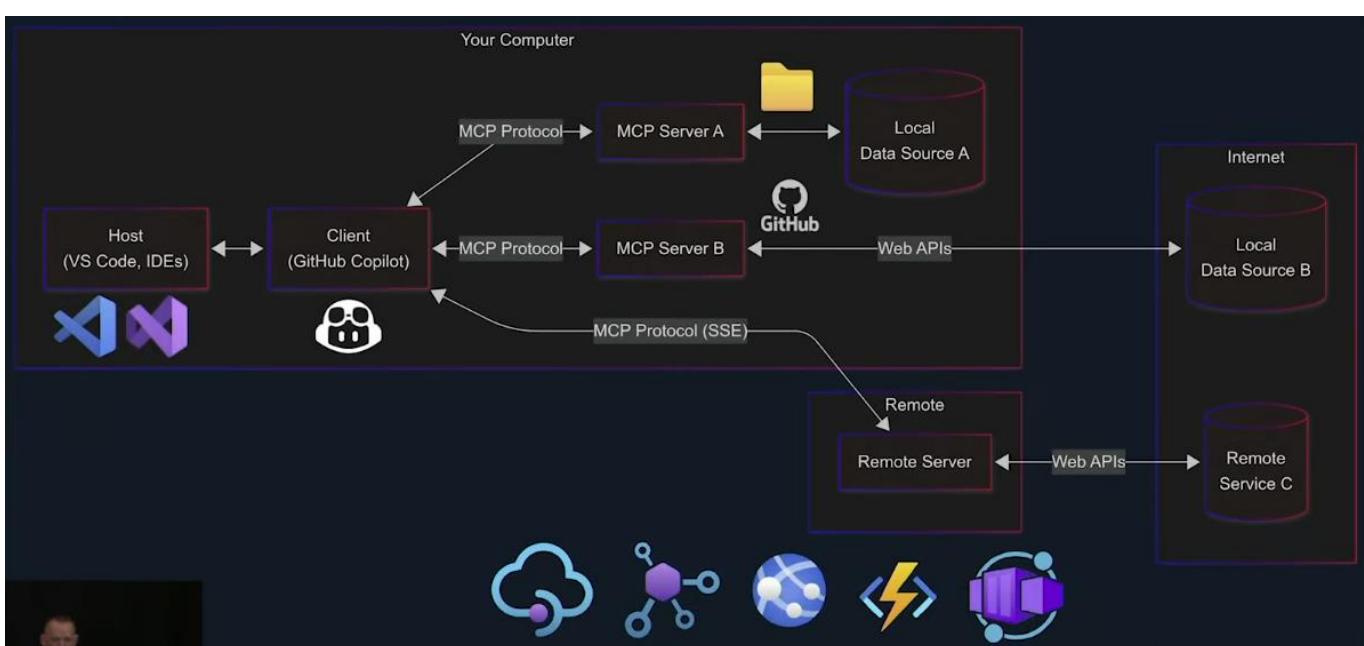
```
 14 updated with any relevant changes, including to project structure
 15 and scripts, and programming guidance
 16
 17 ### Global language guidance
 18 - Use type hints for function parameters and return values for
 19 all languages which support them
 20
 21 ### Python and Flask Patterns
 22 - Use SQLAlchemy models for database interactions
 23 - Use Flask blueprints for organizing routes
 24 - Follow RESTful API design principles
```

```
 25
 26 ### Svelte and Astro Patterns
 27
 28 - Use Svelte for interactive components
 29 - Follow Svelte's reactive programming model
 30 - Create reusable components when functionality is used in
 31 multiple places
 32 - Use Astro for page routing and static content
 33
 34 ### Styling
 35 - Use Tailwind CSS classes for styling
 36 - Maintain dark mode theme throughout the application
 37 - Use rounded corners for UI elements
```

```
1 endpoint.instructions.md 2 copilot-instructions.md x 3 app.py 4 games.py 5 pl ...  
.github > copilot-instructions.md  
  32  
  33 ### Styling  
  34  
  35 - Use Tailwind CSS classes for styling  
  36 - Maintain dark mode theme throughout the application  
  37 - Use rounded corners for UI elements  
  38 - Follow modern UI/UX principles with clean, accessible interfaces  
  39  
  40 ### GitHub Actions workflows  
  41  
  42 - Follow good security practices  
  43 - Make sure to explicitly set the workflow permissions  
  44 - Add comments to document what tasks are being performed
```

```
1 endpoint.instructions.md 2 copilot-instructions.md x 3 app.py 4 games.py 5 pl ...  
.github > copilot-instructions.md  
  45  
  46 ## Scripts  
  47  
  48 - Several scripts exist in the `scripts` folder  
  49 - Use existing scripts to perform tasks rather than performing  
    them manually  
  50 - Existing scripts:  
    - `scripts/setup-env.sh`: Performs installation of all Python  
      and Node dependencies  
    - `scripts/run-server-tests.sh`: Calls setup-env, then runs  
      all Python tests  
    - `scripts/start-app.sh`: Calls setup-env, then starts both  
      backend and frontend servers
```

```
1 endpoint.instructions.md 2 copilot-instructions.md x 3 app.py 4 games.py 5 pl ...  
.github > copilot-instructions.md  
  55 ## Repository Structure  
  56  
  57 - `server/`: Flask backend code  
  58 - `models/`: SQLAlchemy ORM models  
  59 - `routes/`: API endpoints organized by resource  
  60 - `tests/`: Unit tests for the API  
  61 - `utils/`: Utility functions and helpers  
  62 - `client/`: Astro/Svelte frontend code  
  63 - `src/components/`: Reusable Svelte components  
  64 - `src/layouts/`: Astro layout templates  
  65 - `src/pages/`: Astro page routes  
  66 - `src/styles/`: CSS and Tailwind configuration  
  - `scripts/`: Development and deployment scripts
```



EXPLORER

AI-GEEKTRAINER [CODES...]

- .devcontainer
- .github
 - instructions
 - flask-endpoint.instructions.md
 - python-tests.instructions.md
 - ui.instructions.md
 - workflows
 - CODEOWNERS
 - copilot-instructions.md
- dependabot.yml
- vscode

.github > copilot-instructions.md

```
55 ## Repository Structure
56
57 - `server/`: Flask backend code
58 - `models/`: SQLAlchemy ORM models
59 - `routes/`: API endpoints
60 organized by resource
61 - `tests/`: Unit tests for the API
62 - `utils/`: Utility functions and
63 - `client/`: Astro/Svelte frontend
   code
   - `src/components/`: Reusable
     Svelte components
```

flask-endpoint.instructions.md x copilot-instructions.md app.py games.py ...

.github > instructions > flask-endpoint.instructions.md

```
1 # Endpoint creation guidelines
2
3 ## Endpoint notes
4
5 - Endpoints are created in Flask using blueprints
6 - Create a centralized function for accessing data
7 - All endpoints require tests
8   - Use the `unittest` module for testing
9   - All tests must pass
10  - A script is provided to run tests at `scripts/run-server-test`
11
12 ## Project notes
```

flask-endpoint.instructions.md x copilot-instructions.md app.py games.py ...

.github > instructions > flask-endpoint.instructions.md

```
12 ## Project notes
13
14 - The Python virtual environment is located in the root of the project
15 - Register all blueprints in `server/app.py`
16 - Use the [test instructions](.github/instructions/python-tests.in...
```

18 ## Prototype files
19
20 - [Endpoint prototype](server/routes/games.py)
21 - [Tests prototype](server/tests/test_games.py)
22

EXPLORER

AI-GEEKTRAINER [CODES...]

- .devcontainer
- .github
 - instructions
 - flask-endpoint.instructions.md
 - python-tests.instructions.md
 - ui.instructions.md
 - workflows
 - CODEOWNERS
 - copilot-instructions.md
- dependabot.yml
- vscode

.github > instructions > python-tests.instructions.md

```
1 —
2 applyTo: 'server/tests/test_*.py'
3 —
4
5 # Testing notes
6
7 - Tests should create shared data at the top to be used for the tests below
8 - Include tests for success and for data not returned
9 - Use a in-memory SQLite when testing data
10 - Utilize setup and teardown functions to create and destroy the database for testing
11   - Ensure the database is properly closed with `db.engine.dispose()`
```

flask-endpoint.instructions.md python-tests.instructions.md app.py games.py M publishers.py ...

.github > instructions > python-tests.instructions.md

```
1 —
2 applyTo: 'server/tests/test_*.py'
3 —
4
5 # Testing notes
6
7 - Tests should create shared data at the top to be used for the tests below
8 - Include tests for success and for data not returned
9 - Use a in-memory SQLite when testing data
10 - Utilize setup and teardown functions to create and destroy the database for testing
11   - Ensure the database is properly closed with `db.engine.dispose()`
```

A screenshot of a GitHub repository page for 'agents-in-sdlc'. The main navigation bar includes 'Code', 'Issues', 'Pull requests', 'Actions', 'Wiki', 'Security', 'Insights', and 'Settings'. The 'Pull requests' tab is selected, showing a single pull request from 'GeekTrainer' titled 'Finished updates to coding agent scenarios and flow'. The pull request has 27 lines (18 loc) and is 2.13 KB. The commit message is: 'Finished updates to coding agent scenarios and flow'. The pull request was created 5 days ago. The pull request details page shows the title 'Agents in the software development lifecycle (SDLC)' and a note about recent additions to GitHub Copilot. The 'Preview' tab is selected.

A screenshot of a GitHub repository page for 'ai-geektrainer'. The main navigation bar includes 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', and 'Insights'. The 'Pull requests' tab is selected, showing a pull request from 'Copilot' titled 'Add CRUD endpoints for games API with comprehensive error handling and tests #3'. The pull request is marked as a draft and wants to merge 2 commits from 'copilot/fix-2'. The pull request details page shows the title 'Add CRUD endpoints for games API with comprehensive error handling and tests #3' and a note from 'Copilot' about implementing CRUD functionality. The 'Conversation' section shows a comment from 'Copilot' and a list of reviewers: 'GeekTrainer' and 'chrisreddington'.

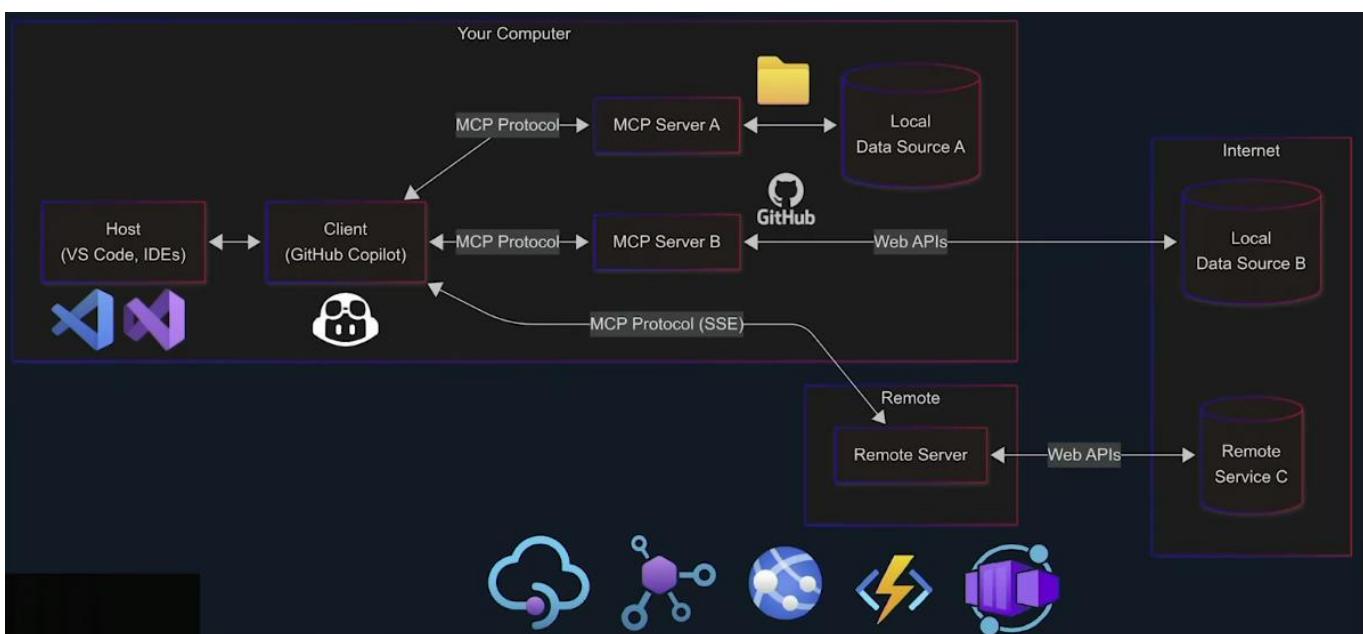
A screenshot of a GitHub repository page for 'ai-geektrainer'. The main navigation bar includes 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', and 'Insights'. The 'Pull requests' tab is selected, showing a pull request from 'Copilot' titled 'Add CRUD endpoints for games API with comprehensive error handling and tests #3'. The pull request is marked as a draft and wants to merge 2 commits from 'copilot/fix-2'. The pull request details page shows the title 'Add CRUD endpoints for games API with comprehensive error handling and tests #3' and a note from 'Copilot' about implementing CRUD functionality. The 'Assignees' section is open, showing a search bar and a list of suggestions: 'Copilot' (selected), 'GeekTrainer', 'chrisreddington', 'Damovisa', 'joshjohanning', and 'LadyKerr'. The 'Assignees' section also includes a note: 'Assign up to 10 people to this pull request'.

docs > 1-copilot-coding-agent.md > # Exercise 1 - GitHub Copilot coding agent > ## Resources

```

1  # Exercise 1 - GitHub Copilot coding agent
123 ## Create new endpoints to modify games
132 6. Set the **Description** to:
133
134
135      ``markdown
136      We're going to be creating functionality in the future to allow for the
137      submission (and editing) of games. For now we just want the endpoints so we
138      can explore how we want to create the UX and do some acceptance testing. Our
139      requirements are:
140
141      - Add new endpoints to the Games API to support creating, updating and
142      deleting games
143      - There should be appropriate error handling for all new endpoints
144      - There should be unit tests created for all new endpoints
145      - Before creating the PR, ensure all tests pass
146
147      Note the level of guidance being provided to Copilot to help set everyone up
148      for success.
149      Scroll towards the bottom of the dialog box to locate the *Assignments* button

```



Add CRU endpoints for games API with comprehensive error handling and tests #3
Copilot wants to merge 2 commits into `main` from `copilot/fix-2`

This PR implements the complete CRU (Create, Read, Update, Delete) functionality for the games API as requested by the design team for UX development and acceptance testing.

New Endpoints Added

POST /api/games

Creates a new game with validation for all required fields:

```
{
  "title": "New Game Title",
  "description": "Game description (min 10 chars)",
  "publisher_id": 1,
  "category_id": 1,
  "star_rating": 4.5
}
```

Assignees: GeekTrainer, chrisreddington

Labels: None yet

Projects: None yet

Milestone: None

This branch is 3 commits ahead of [main](#).

Copilot and GeekTrainer Add search functionality to game... 4d056ac · 15 minutes ago

.devcontainer Initial commit 6 hours ago

.devcontainer Initial commit 6 hours ago

About

No description, website, or topics provided.

- Readme
- MIT license
- Code of conduct
- Security policy
- Activity
- Custom properties

This branch is 3 commits ahead of

Copilot and GeekTrainer Add search

.devcontainer

.github

Local Codespaces

Clone

HTTPS SSH GitHub CLI

<https://github.com/octohol/ai-geektrainer>

Clone using the web URL.

Open with GitHub Desktop

Download ZIP

About

No description, website, or topics provided.

- Readme
- MIT license
- Code of conduct
- Security policy
- Activity
- Custom properties
- 0 stars

.github > instructions > python-tests.instructions.md

1 —

2 applyTo: 'server/tests/test_*.py'

3 —

4

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS 3 ...

@GeekTrainer → /workspaces/ai-geektrainer (main) \$ git checkout copilot/fix-2
branch 'copilot/fix-2' set up to track 'origin/copilot/fix-2'.
Switched to a new branch 'copilot/fix-2'

@GeekTrainer → /workspaces/ai-geektrainer (copilot/fix-2) \$

The screenshot shows the VS Code interface with the MCP configuration file open in the editor. The file path is `.vscode > mcp.json`. The code defines inputs for GitHub Personal Access Tokens and a server setup for GitHub using Docker.

```
1 {
2     "inputs": [
3         {
4             "type": "promptString",
5             "id": "github_token",
6             "description": "GitHub Personal Access Token"
7             "password": true
8         }
9     ],
10    "servers": {
11        "github": {
12            "command": "docker",
13            "args": [
14                "run",
15                "-i",
16                "--rm",
17                "-e",
18                "GITHUB_PERSONAL_ACCESS_TOKEN"
19            ]
20        }
21    }
22}
```