

Lab 087: Copilot + MCP : GitHub, Playwright et votre propre serveur

Level: L200 Parcours: copilot Durée: ~60 min Coût: GitHub Free

info "Conception gratuite d'abord" Terminez l'exercice avec Copilot Free ou la simulation incluse. Les fonctions payantes sont facultatives et clairement indiquées.

tip "Impact sur les AI Credits" Faible pour l'exercice guidé ; moyen/élevé pour les exécutions agentiques ou répétées. Préférez la sélection automatique du modèle, planifiez avant d'agir, gardez un contexte ciblé et évitez les sessions parallèles inutiles.

Ce que vous apprendrez

Connecter des outils MCP approuvés à Copilot Comparer les serveurs GitHub, Playwright et local Appliquer listes d'autorisation et indications lecture seule Concevoir un contrat d'outil minimal

Alignement GH-300

Use Copilot features (25?30%) · Data and architecture (10?15%). Cette correspondance suit l'instantané GH-300 du 07/08/2026. Le plan officiel peut conserver des termes anciens comme « Copilot Edits » ; ce lab enseigne l'expérience actuelle et signale la différence.

Carte conceptuelle

[diagram or code sample]

Exercice pratique

Étape 1: Inventorier les serveurs

Listez serveur, transport, identifiants, frontière de données et outils avant connexion.

Préférez le registre MCP GitHub ou un serveur inspectable.

Étape 2: Exécuter une tâche GitHub en lecture seule

Demandez à Copilot de lister des issues ou métadonnées sans modifier l'état. Confirmez chaque appel avant d'élargir les droits.

Étape 3: Tester Playwright en local

Utilisez Playwright MCP uniquement sur localhost. Relevez un point d'accessibilité et n'envoyez jamais de données privées à un serveur non vérifié.

Étape 4: Réutiliser le Lab 020

Exposez un outil local OutdoorGear. Ajoutez schéma strict, sortie bornée, erreurs explicites et liste d'autorisation.

Kit d'accompagnement

Le build du site génère des téléchargements PDF et ZIP stables. Le ZIP contient le guide du lab et tous les fichiers d'accompagnement déclarés.

Télécharger le PDF généré Télécharger le ZIP généré

Supporting file: mcp.json

```
{ "servers": { "outdoorgear": { "command": "python", "args": [ "server.py" ] } } }
```

Supporting file: mcp_allowlist.json

```
{ "allowed_servers": [ "outdoorgear" ], "allowed_tools": [ "catalog_search" ] }
```

Supporting file: threat_checklist.csv

threat,control,evidence untrusted-tool-output,schema validation,negative test
excess-permission,allowlist,approved server list