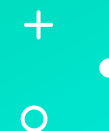


Curso: VSCode y GitHub Copilot



Tema 2: Chat Agent

○

- **CONNECT**
 - *¿Qué hace diferente al chat?*
 - La conversación continuada y guardable
 - *¿Qué modos tenemos?*
 - 1 o 2 inútiles y El Agente
 - Configuración
 - La carpeta .github



CONCEPTS:

Prompts ,Instructions, Tools

Prompts: What?

- Comandos Slash 🎸
- Oficiales: `/clear /doc /explain /fix /new /test`
- Biblioteca de peticiones reutilizables:
 - en la carpeta `.github/prompts`
 - `[command].prompt.md`
- Enriquecidos con:
 - `#` : `#file #selection`
 - `@` : `@workspace @vscode @terminal @github`

Instructions: How?

- Generales en fichero
 - `.github/copilot-instructions.md`
 - `AGENTS.md`
- Contextuales:
 - en la carpeta `.github/instructions`
 - `[context].instructions.md`
- Una cabecera en Front-matter que indica:
 - propósito y
 - ficheros a los que aplica.

Tools

- Herramientas externas en lenguaje natural
- Dos ejemplos arquetípicos:
 - local : Manipulación de **Playwright**
 - remote : API de **GitHub**
- Cada MCP dispone de un server con múltiples Tools
- Puede requerir autenticación y configuración extra.

CONCEPTUAL DEMO

Show me the code;

- ❑ Configure: instructions

- ❑ Add: weather report spec

- ❑ Git: commit prompt

- ❑ *Bonus...Clean: refactor HttpClient*

-

Context:

- The functional PRD
- The current Rest API

Workflow :

1. Create or copy instructions files to guide the LLM code generation.
2. Write the spec for the RF2
3. Implement the code
4. Fix if needed
5. Commit the changes

Validation :

- ✓ RF-2: `GET /activities/:id`
 - ✓ Details of a specific activity
 - ✓ 404 when activity not found
- ✓ Bonus:
 - ✓ 400 when id malformed

CONCRETE PRACTICE

Activity-Bookings

CONCLUSION

- *¿Qué hemos aprendido?*
- ×
 - El modo agente será tu pair programmer
 - Configurar con instrucciones
 - Prompts as code
- *¿Qué haremos mañana?*
- Verificar el código