

QF-TEST



Das Tool für professionelle UI-Testautomatisierung
Desktop, Web & mobile Anwendungen einfach und effizient testen

QF-Test Spezialwebinar

KI-Agenten im Einsatz

Neue Effizienz mit QF-Test 11 durch KI und MCP

Daniel Rieth

Max Melzer

Dr. Pascal Bihler

17.08.2026

Agenda

1. **KI und QF-Test**
2. **KI-Modelle verbinden**
3. **KI-Datengenerierung**
4. **KI-Generierung von Testsuiten**
5. **Der KI-Anweisung Knoten**
6. **QF-Test und MCP**

KI in QF-Test



- **Tests explorativ oder aus Dokumentation generieren**
- **KI-Anweisungen und Entscheidungen in Tests integrieren**
- **„Bring your own model“**

KI mit QF-Test



- **Steuern von QF-Test und Drittanwendungen mit Claude Code etc.**
- **MCP-Baukasten: Beliebige Anwendungen fit für KI machen**

Agenda

1. KI und QF-Test
2. KI-Modelle verbinden
3. KI-Generierung von Testsuiten
4. Der KI-Anweisung Knoten
5. KI-Datengenerierung
6. QF-Test und MCP

KI Modelle verbinden

Optionen

QF-Test Optionen

Suche nach Optionen

> Allgemein

> Test-Management

> Aufnahme

> Wiedergabe

SmartID und qfs:label

Android

iOS

> Web

SWT

UI-Inspektor

> Künstliche Intelligenz

MCP & Agentenfähigkeiten

KI-Sicherheit

"Large Language Model"-Integration

☐ KI-Funktionen blockieren

Blockieren bedeutet, dass neue oder bestehende KI-Funktionen in QF-Test nicht genutzt werden können.
Blockiert werden: KI-Suitegenerierung, KI-Anweisung, Text-Check mit KI, der MCP-Server und das AI-Skripting-Modul.

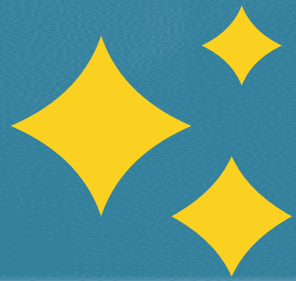
+ ✎ ✕ ⬆ ⬇

KI-Konfigurationen

Type	Name	URL oder Pfad	API-Schlüssel	Modellname
Anthropic	default	https://ai-gateway-example-url.com/v1	sEcr3t4p1K3y	chat/default
Ollama	localAI	http://localhost:12345		glm-4.7-flash
Claude Code CLI	claude	C:\Users\daniel\local\bin\claude.exe		claude-sonnet-5
Github Copilot ...	github			

Agenda

1. KI und QF-Test
2. KI-Modelle verbinden
3. KI-Datengenerierung
4. KI-Generierung von Testsuiten
5. Der KI-Anweisung Knoten
6. QF-Test und MCP



KI-Datengenerierung

- **KI erstellt Tabellendaten nach Prompt**
- **@mcp/prompts - überschreibbar**



qfs-ai.qft

Beinhaltet @mcp/prompts, @mcp/tools und Hilfsmethoden

Überschreibbar in eigenen Suiten (Optionen > "MCP & Agentenfähigkeiten")

Agenda

1. KI und QF-Test
2. KI-Modelle verbinden
3. KI-Datengenerierung
4. KI-Generierung von Testsuiten
5. Der KI-Anweisung Knoten
6. QF-Test und MCP



KI-Generierung von Testsuiten

- generiert Testentwurf
(verwendet @mcp/prompt "Generate Suite")
- exploriert selbstständig die Anwendung
(nutzt QFS-MCP Tools)

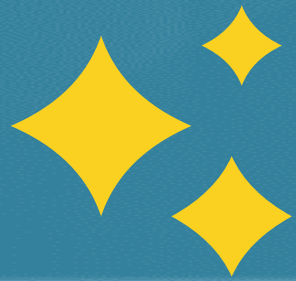


KI-Generierung von Testsuiten

DEMO

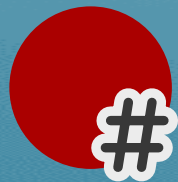
Agenda

1. KI und QF-Test
2. KI-Modelle verbinden
3. KI-Datengenerierung
4. KI-Generierung von Testsuiten
5. Der KI-Anweisung Knoten
6. QF-Test und MCP



KI-Anweisung Knoten

- führt Prompt aus unter Verwendung von QFS-MCP Tools
- Konvertierung durch Aktivierung der Aufnahme vor der Ausführung





KI-Anweisung Knoten

DEMO

Agenda

1. KI und QF-Test
2. KI-Modelle verbinden
3. KI-Datengenerierung
4. KI-Generierung von Testsuiten
5. Der KI-Anweisung Knoten
6. QF-Test und MCP



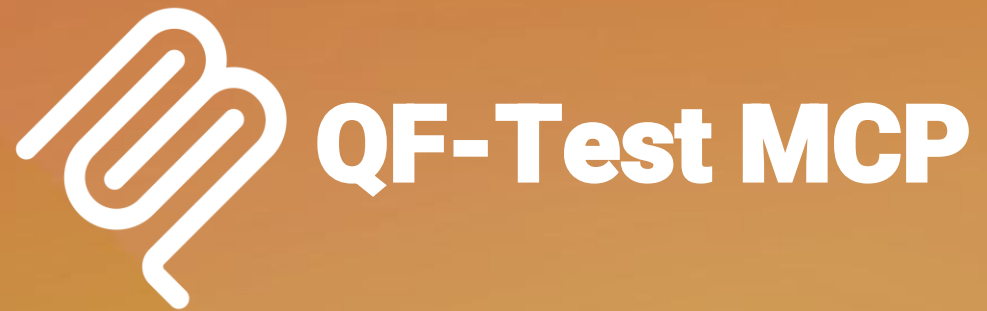
QF-Test MCP

- **Tools für QFS-MCP-Server**
- **Tools als QF-Test Prozeduren selbst definierbar (@mcp/tools)**
- **Freigabe für internen/externen Gebrauch in Optionen ("KI-Sicherheit")**



Annotationen für eigene Tools/Prompts

Annotation	Bedeutung
@mcp/prompts	Prozeduren in Package sind Prompts
@mcp/tools	Prozeduren in Package sind Tools
@mcp/name abc	Überschreibt Standardnamen von Prompts/Tools
@param abc	Parameter, der in Tool verwendet wird (mandatorisch, wenn Beschreibung mit "(Required)" beginnt)
@param reason	KI muss Grund angeben, warum Tool benutzt wird (Logging)
@return	Rückgabewert des Tools
@nodebug	Debugging für diesen Knoten deaktivieren



DEMO

Fazit

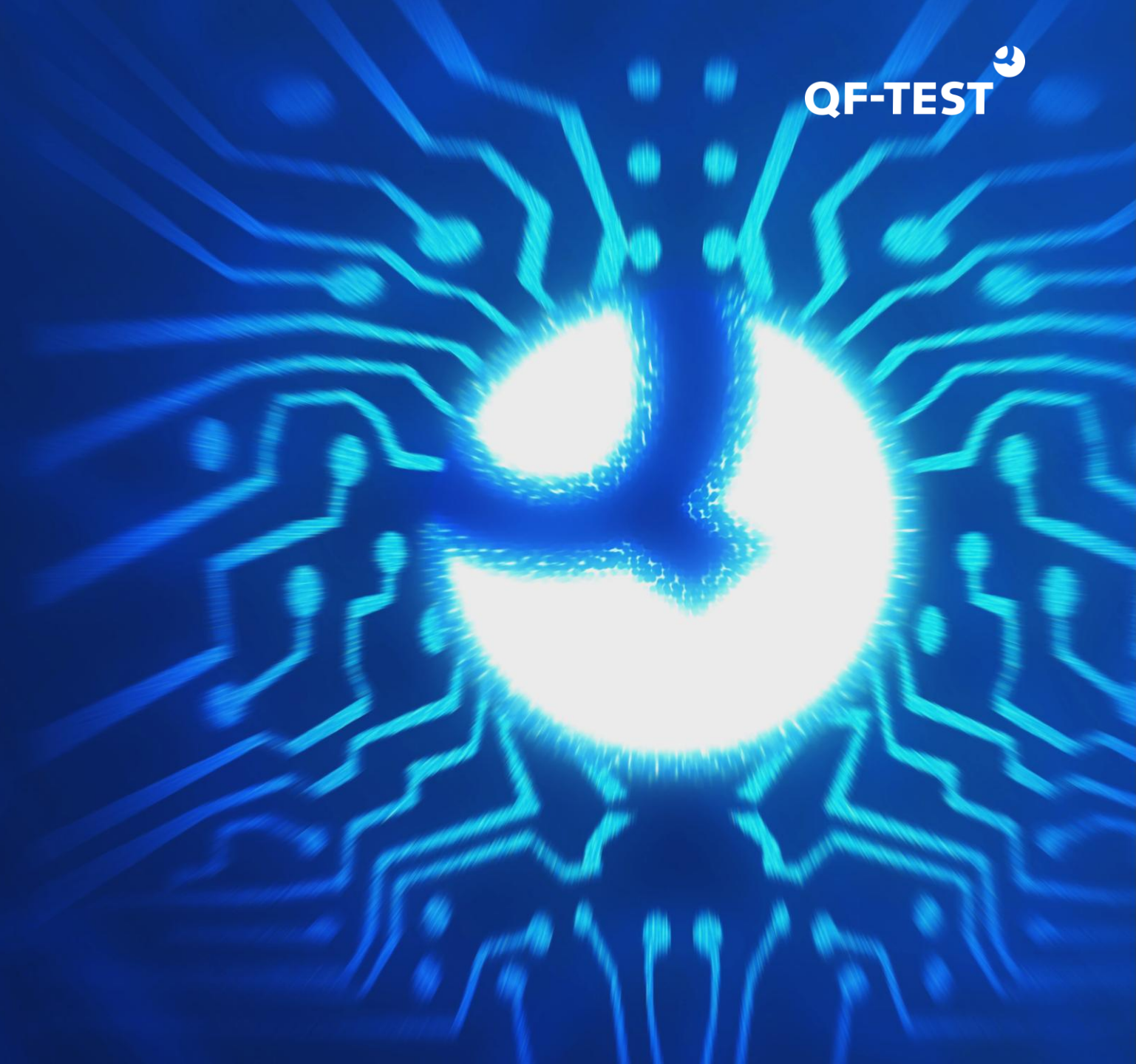
QF-Test 11 bringt mächtige KI-Werkzeuge zur Testgenerierung und -Ausführung.



Der innovative MCP-Baukasten ermöglicht, bisher nicht KI-fähige Anwendungen in agentische Workflows einzubinden.



DANKE!



Q&A

