

Web Autopilot

Enterprise-Grade Testing for Copilot-Style LLM Applications

The Problem: You can't test if an LLM's answer is "correct" - it's non-deterministic.

The Solution: Test *structure*, not *semantics*. Validate JSON format, response timing, and error handling.

FOUR TESTING PILLARS

1. Structure Over Semantics

- JSON validity when JSON requested
- Code block syntax validation
- Markdown table/list formatting
- Response non-empty and timely

2. Time to First Token (TTFT)

- Critical streaming UX metric
- Target: < 3 seconds
- Automated trend tracking
- Regression alerting

3. Chaos Exploration

- Autonomous "monkey" testing
- Seeded PRNG for reproducibility
- Finds edge cases humans miss
- Safety: never clicks Delete/Logout

4. Reliability Injection

- Offline handling verification
- 429/500/503 error responses
- Slow/flaky network simulation
- Graceful degradation checks

MODULES INCLUDED

Module	Purpose	Key Capability
ChaosRunner	Random exploration	Reproducible with --seed 12345
StreamingValidator	Streaming response testing	TTFT measurement, stop button verification
PromptCorpusTester	Response format validation	JSON, code, lists, tables without semantic checks
NetworkInjector	Failure simulation	Offline, slow, 429, 500, 503 injection
PerformanceMetrics	Performance tracking	TTFT trends, memory leaks, scroll FPS
JsonValidator	JSON response validation	Auto-repair, field checking, extraction

COPILOT APPLICATIONS

M365 Copilot

- Streaming render in Word/Excel/PPT
- Format preservation validation
- Offline graceful degradation

Teams/GitHub Copilot

- Multi-turn conversation state
- Code block syntax validation
- Suggestion rendering speed

ENTERPRISE VALUE

Without Web Autopilot	With Web Autopilot
Manual testing, can't scale	Automated CI/CD integration
"Can't test LLM apps"	Structure + reliability testing
No performance baseline	TTFT trends + alerting

Edge cases found in production

Chaos runner finds them first

"Works on my machine"

Reproducible with seeds

GitHub: github.com/arnavmmittal/web-autopilot

Stack: TypeScript, Playwright, Node.js

License: MIT