

Maximum Retry Settings for Scenario Editor Tests

Recommended guidance

Use retry settings to reduce noise from temporary timing or environment issues, not to hide unstable locators, incorrect waits, broken test data, or real application defects. A retry should make a valid test more resilient; it should not turn a failing test into an unclear result.

Good use cases

- Short-lived network delays during page load or service calls.
- Minor timing variation in asynchronous UI behavior.
- Known non-deterministic behavior in a test environment that is already being tracked.

Use caution when

- The same step fails repeatedly across retries.
- The retry makes execution time too long for CI/CD or nightly runs.
- The failure involves data creation, purchases, messages, or any action that should not be repeated blindly.
- The scenario lacks clear assertions after the retried action.

Recommended workflow

1. Fix selectors, waits, and test data before increasing retries.
2. Set the smallest retry count that addresses the known transient issue.
3. Document why the retry exists and what condition should allow it to be removed.
4. Review retry-heavy scenarios during maintenance because they often indicate hidden instability.

Last updated: October 5, 2026

Revision #3

Created 2026-07-31 15:39:45 UTC by Admin

Updated 2026-10-05 17:36:26 UTC by mmora