



Live Preview

Overview

Live preview gives the real-time execution result for a [test case](#). This allows testers to observe the execution of test steps and their impact on the application under test. The live preview helps detect potential errors or issues before running the full test case, saving time and effort. These real-time results can only be viewed once and are not accessible under the [results](#) page, where the test results are usually stored.

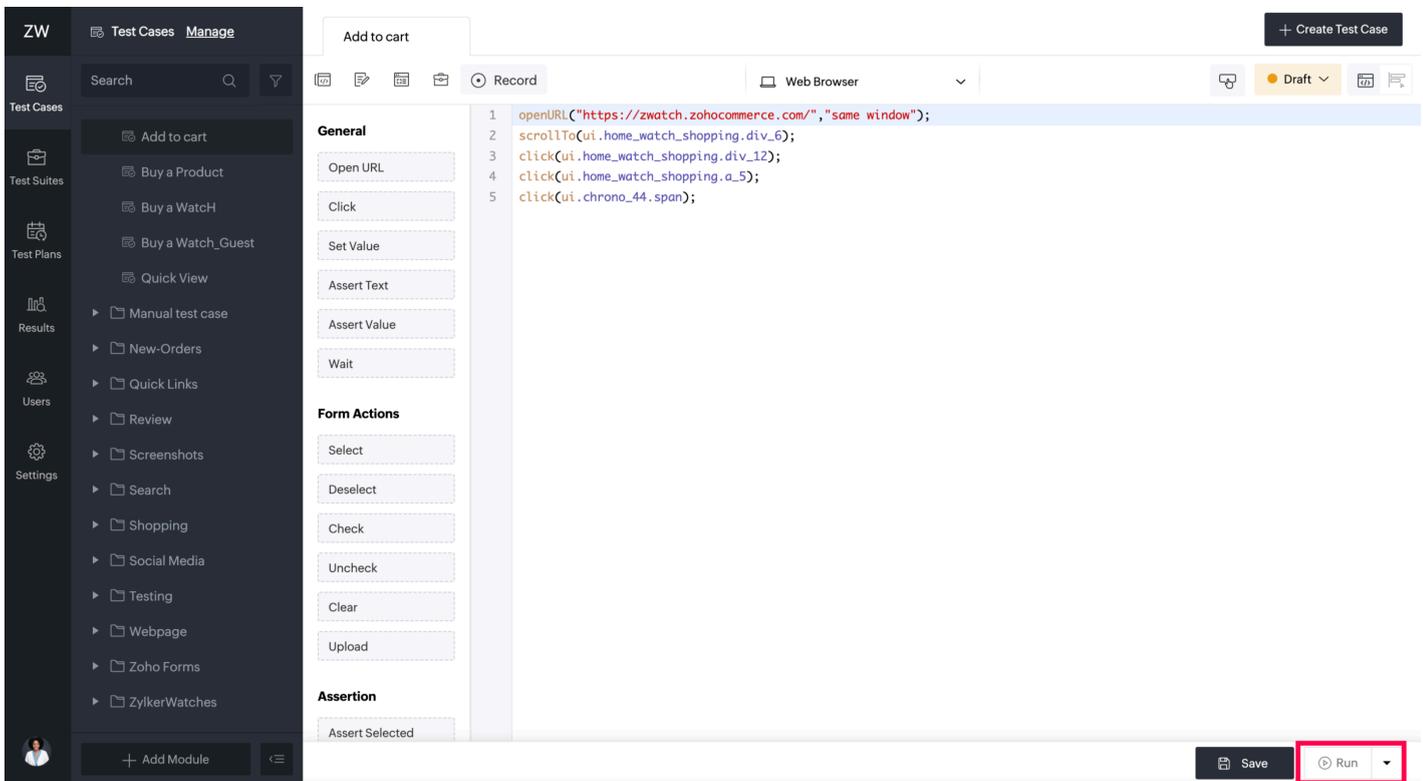
To execute and preview test cases, configure a run using either of the two agents, Zoho Cloud or a [local agent](#), and specify the browser and screen configurations required.

Additionally, you can choose to run with the [environment variable](#), in case the test case needs to be run under specific environments.

Live preview for web

To preview the real-time execution of web test cases, customize the run configuration using the steps below:

1. Click **Run** on a test case to initiate a *Preview Run*.



Note: In order to run a live preview, the test case has to be saved.

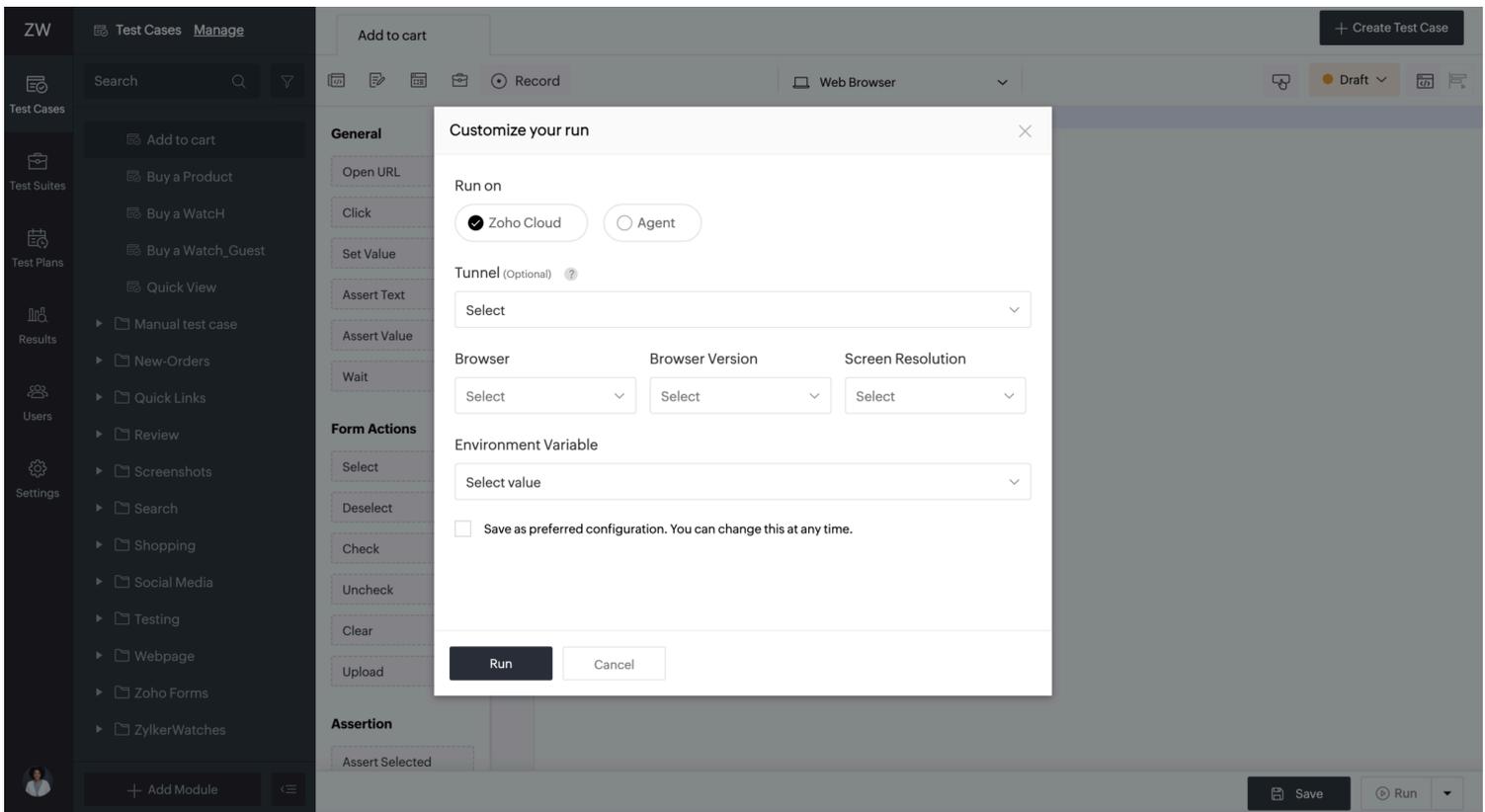
2. Select the desired *Agent*, *Environment*, and *Screen Resolution*.

Agent: Zoho QEngine provides two kinds of agents, Zoho Cloud and Agent. By default, the Zoho Cloud agent will be selected.

- Using the Zoho Cloud agent, you can choose the browser, its version, and the desired screen resolution for the test preview run.
- To use the local Agent, the agent extension should be installed on your machine. [Learn more about Agent installation here](#). The browser and its version are selected based on the environment and the version of the browser currently available on your machine.

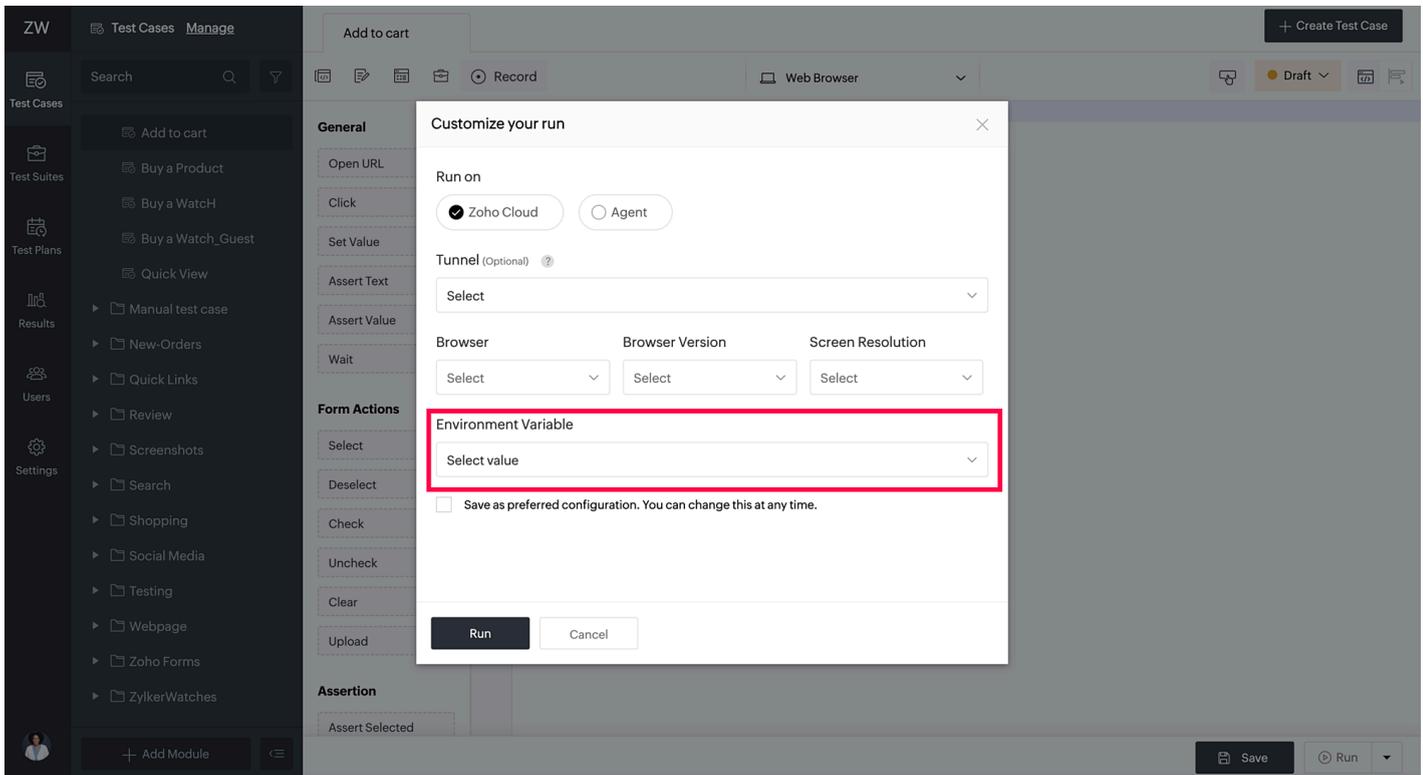
Environment: Select the browser and its version used for the device used during test case execution.

Screen resolution: Specify the screen resolution in pixels for the device used during the test case execution.



Note: During *Run* you will see the testing steps live in action if you select browsers like, Chrome, Firefox, Safari and others. However, if you opt for the headless execution, the graphical user interface (UI) of the browsers will not be visible. Nevertheless, the test case will still run, and the results will be displayed.

3. Select the desired environment variable from the dropdown, if you wish to test with them. If you don't have anything chosen, the [global variable](#) will be considered by default.



4. Check *Save as preferred configuration* to avoid configuring the run every time. If not, Zoho QEngine will ask for your choice every time you click **Run**.

 **Note:** If you set a preferred configuration, click the dropdown next to *Run*, then click **Customize** to change the configurations again.

5. Once you choose the required configuration, click **Run**.

Live preview summary

The run preview page comprises details such as logs, details of the agent, device and screen resolution used, and other details of the test case execution.

Let's take a detailed look at what each section below represents:

The screenshot displays the Zoho QEngine interface for a 'Preview Run'. The left sidebar contains navigation options like 'Test Cases', 'Test Suites', 'Test Plans', 'Results', 'Users', and 'Settings'. The main content area shows the 'Preview Run' summary with the following details:

- RUN SUMMARY 1**: Results: Passed, Duration: 14.09s, Version: V 3.6, Agent Name: Ann Agent.
- PLATFORM 2**: Chrome (Headless), 114, 1440x900 Px.
- PREVIEW RUN 3**: Includes a 'Restart' button.
- Logs 4**: A table of execution logs with columns for step number, description, and duration.
- SCREENSHOTS 8**: A preview of the web page being tested, showing a product page for a watch.

• Run summary

This section gives an overview of the run's overall status, including whether the test case was stopped, passed, or failed. It also displays the test execution duration, the version of the test case run, and the selected agent for the run.

Note: If a test preview summary gives a passed test case, the *Duration* in preview would be the overall execution time. But, if the test case fails in a test preview. For example, if there is an element locator not currently visible in the DOM, Zoho QEngine waits for 60 seconds to find the locator. If the issue persists, the execution of the test case will be terminated.

• Platform

This section gives the details about the operating system, browser, and the screen resolution chosen for the run.

- **OS** - The operating system chosen for the run.

- **Browser** - The name and the version of the browser chosen for the run.
- **Resolution** - The screen resolution chosen for the run.
- **Restart**

This restarts the preview run for the same configuration chosen during the original run.
- **Logs**

They show the individual test steps that are successfully executed or failed.
- **Console log**

This displays real-time warnings and errors for failed test cases, providing valuable debugging information necessary to troubleshoot and improve the execution of a test script.

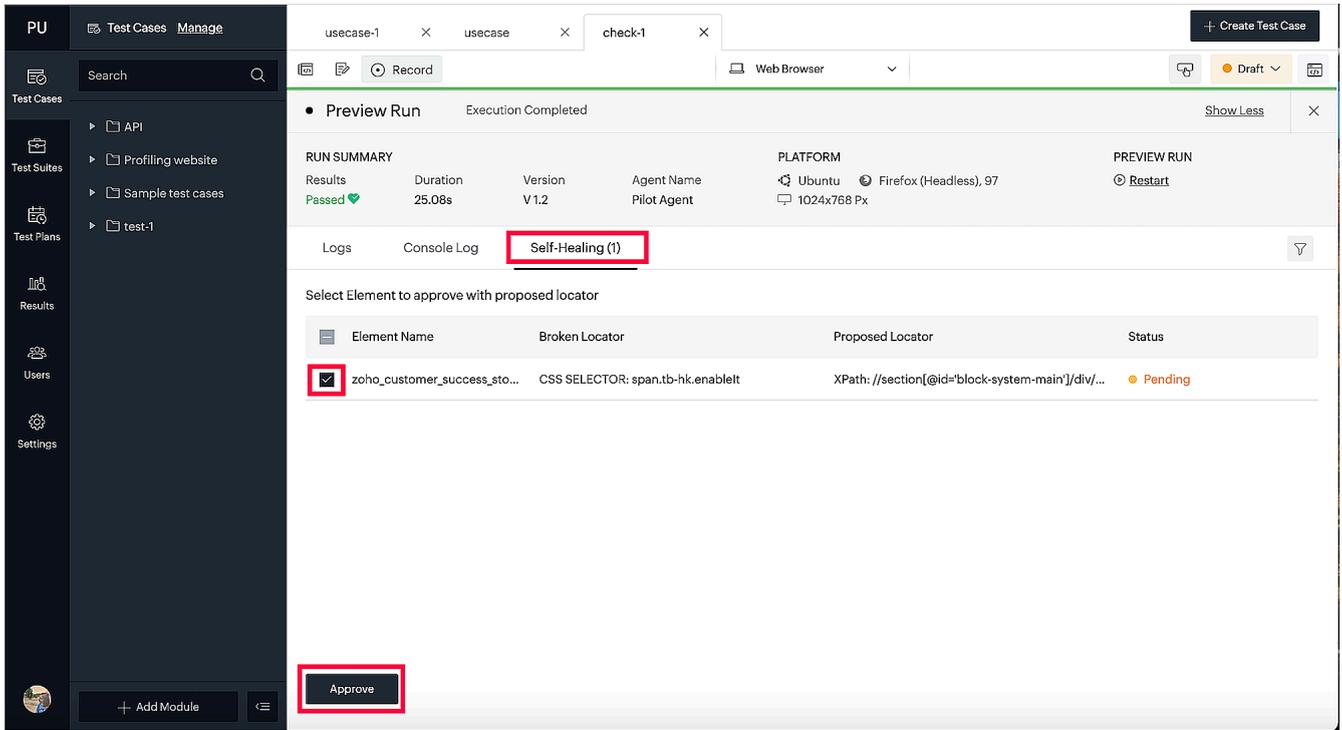
 **Note:** Test cases experience failures as a result of console log errors or warnings, but this occurs only when these issues impact the execution of subsequent steps.

- **Self-healing**

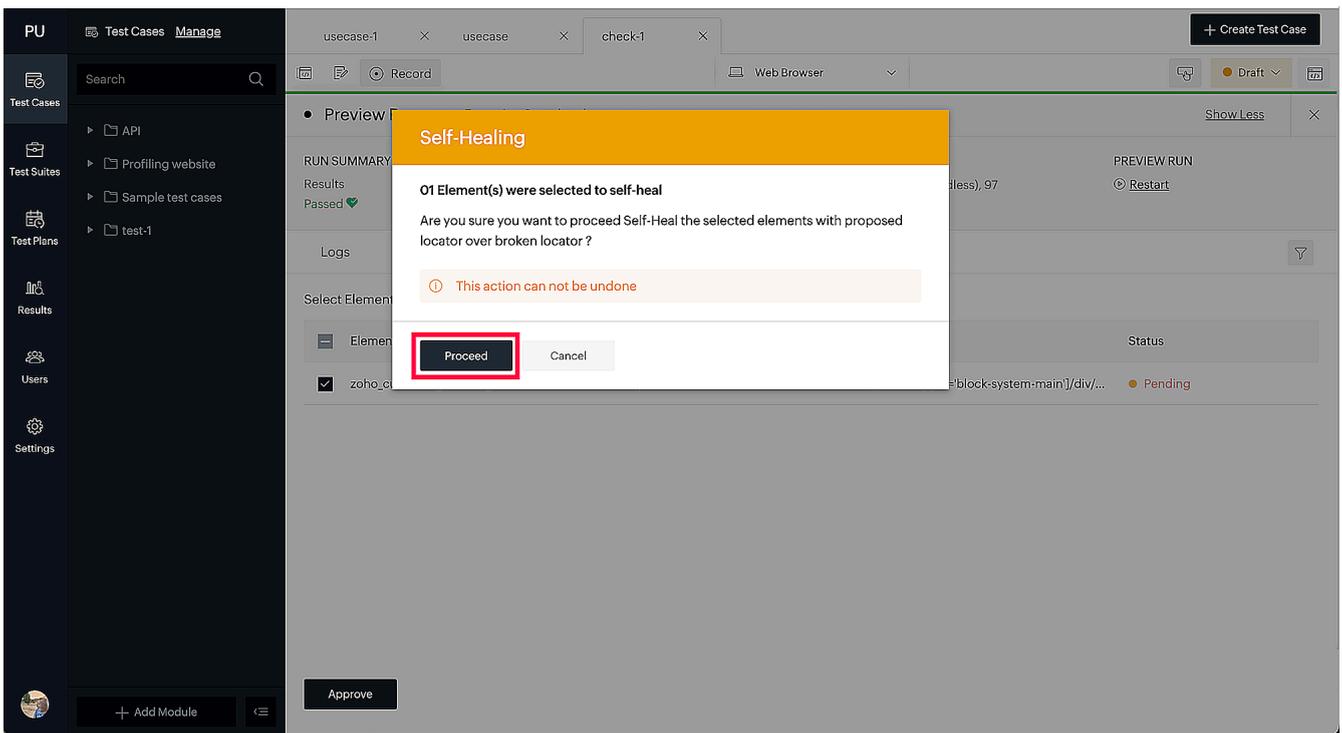
This is a mechanism Zoho QEngine uses to stabilize the test cases over broken locators. When recording interacted elements, multiple locators are captured for each element, which are then assigned an order of priority upon saving. Enabling self-healing allows broken locators to be replaced with the next available locator from the prioritized list. The order of locators can be easily modified at any time by rearranging them as needed.

 **Note:** For this feature, you have to enable *Self-Healing* in the *Preferences* section of the settings.

After test case execution, the self-healing logs will display a list of locators that require repair. If you choose to repair the locators, you can approve the requests. Simply select the broken locators you wish to repair and click **Approve**. Subsequently, the broken locator will be replaced with the proposed locator in future executions.

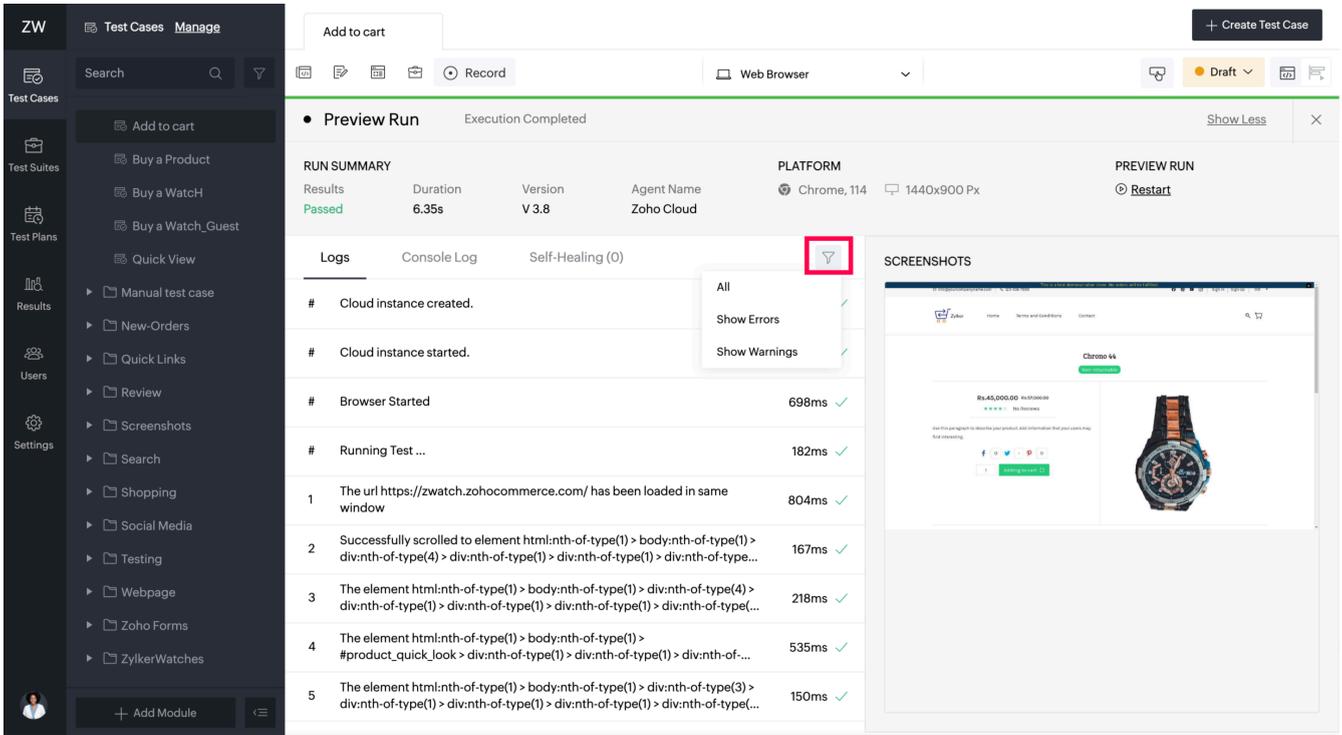


Once you approve that, there will be a pop-up asking you to confirm the request. Click **Proceed**.



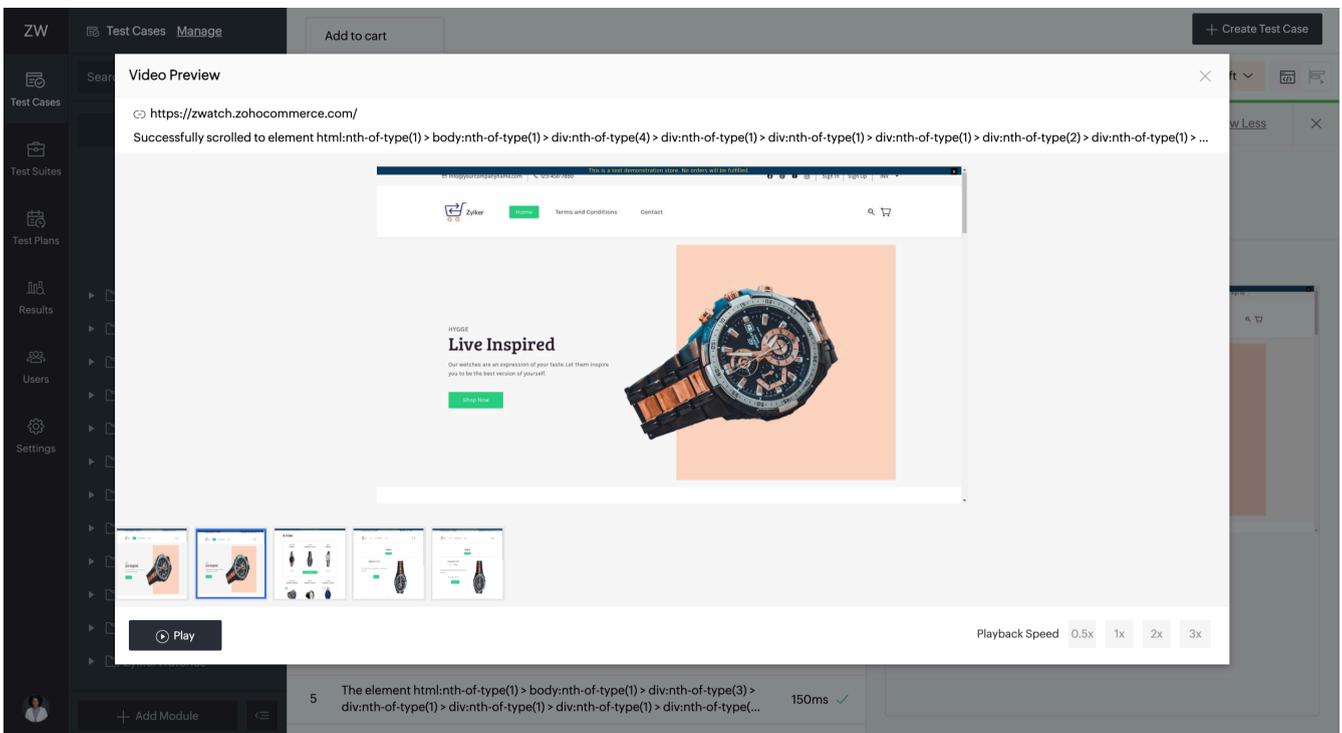
- **Filter logs**

You can filter logs for errors or warnings by filtering them using the options, *All*, *Show Errors*, and *Show Warnings*.



• Screenshots

This section displays the preview (visual navigation of the screens accessed and actions performed on the web). You can either play this as a video or navigate through the various images using the "Next" and "Previous" controls. There is also an option to replay the video preview at specific playback speeds (0.5x, 1x, 2x, 3x), too.



 **Note:** If you click **Show Less**, the test case editor screen comes up. The preview screen minimizes to the bottom, showing the percentage of the completion (100%), the test step under execution (#15), and mentions the final test step executed. Click **Show More** to bring the live preview results back on screen.

[Live Preview for API](#)

The response section for API test cases is discussed in this section.

Best practices

- Ensure the test case is saved before execution to avoid losing any changes made during test creation.
- Before running a test case, it needs to be checked for correctness and tested for functionality.
- Prior to executing the test case, it must be checked for syntactical errors, missing argument numbers for tasks, argument type, component (variables, elements, functions, tasks, and the like.) availability, correctness, and similar things.
- All updates to the test case are saved. When a test case is run with unsaved changes, it throws an error. For example, The *handleAlert* task requires arguments. When they are not passed, the test case is not allowed to be saved.
- All tasks used follow syntactic notations. The editor identifies the invalid characters and doesn't allow you to save the test case.