

Run Automation Tests

Learn how to run automation tests.



Running Tests from Editor Interface

1. Open your project.
2. Enable the **Functional Testing Editor** and any other plugins containing tests you want to run.
3. Restart Unreal Editor.
4. In the Editor, go to **Tools > Test Automation**.
5. In the **Automation** tab of the **Sessions Frontend** under the **Test** column, select the tests you want to run. You can run an entire section by selecting the parent checkbox. For granular selection, expand each section.

6. In the **Automation** tab toolbar, click the **Start Tests** button.

You can monitor the progress and results in the **Test** panel. Selecting a test row displays the events in the **Results** panel. For screenshot comparison results, use the **Screen Comparison** tab.

Running Tests from Session Frontend

By default, the tests run from an Editor instance. However, the Session Frontend can connect to additional Editor and Client instances. This is useful for network testing.

Before connecting, the additional instances must be running. You can do this by using the Quick Launch feature.

Once running, you can select the instance from the Session Frontend in the left panel under **My Sessions**. The test list will update with compatible tests for the instance.



You can set a test to be Client-only and make it unavailable on Editor instances.

Running Tests from Unreal Frontend

You can launch Client tests as a standalone application from the [Unreal Frontend \(UFE\)](#). This is useful if you want to build, package, and deploy on a remote console without the Editor.

The interface in the **Automation** tab is the same as the Editor.

Running Tests from Command Line

You can run automated tests from the command line by using arguments with the Editor or Client executable, including the following examples:

- `-ExecCmds="Automation RunTest Test1+Test2;Quit"` - Runs `Test1` and `Test2` and then closes the executable.
- `-ExecCmds="Automation RunTest MySet.MySubSet;Quit"` - Runs all tests under `MySet.MySubSet` section.
- `-ExecCmds="Automation RunTest Group:MyGroup;Quit"` - Run all tests filtered in the group `MyGroup`. See [Configure Automation Tests](#) for more information on test groups.
- `-ReportExportPath="<output path>"` - Stores results in a JSON format with related HTML files. See [Setting Up an Automation Test Report Server](#) for more information on HTML reports.
- `-ResumeRunTest` - When used with `-ReportExportPath`, reads the JSON file and resumes the test from the first test marked as not-run. In-progress tests will be marked as a failure. This is useful to resume testing on critical failure.