

| Comparison Criterion | Playwright (Open-Source) | MagicPod (Low-Code/Keyword-Driven/AI) | Testsigma (Low-Code/NLP) |
|-----------------------------------------|--------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| I. CORE OBJECTIVES | | | |
| 1. Test Scripting Method | Code-Driven. Requires writing code in JavaScript/TypeScript/Python. | Low-Code/No-Code. Testers use Keywords (Natural Language) via a GUI to create scenarios. | Low-Code/NLP (Uses Simple Natural Language). Testers type English testing scripts as inputs. |
| 2. Who can use? | Developer (Coding skills mandatory). | Both Developer and Tester (No coding skills required). | Both Developer and Tester (No coding skills required). |
| 3. CI/CD Integration | Good. Requires custom configuration and scripting (Jenkins, GitHub Actions, etc.). | Excellent. Features built-in Webhooks for easy triggering by CI/CD systems. | Excellent. Native integrations and dedicated CLI/Webhooks for seamless CI/CD execution. |
| II. COST & FEASIBILITY | | | |
| 4. Tool Cost | FREE. | HIGH. Starting price is around \$320 - \$400+ / month (excluding add-ons). | Paid Subscription. (more flexible than MagicPod, pending confirmation). |
| 5. Initial Setup Complexity | Fast (simple library installation). | Complex and time-consuming. Requires Agent installation and initial project configuration. | Relatively Fast. (Cloud-based platform, minimal complex local setup required). |
| III. TECHNICAL & AI FEATURES | | | |
| 6. Maintenance & Stability | Low. Requires manual code fixes every time the UI/Locators change (High maintenance cost). | High (AI Locator). AI automatically finds elements whose IDs/Classes have changed, reducing maintenance time by up to 90%. | HIGH. Uses robust AI Self-Healing, comparable to MagicPod. |
| 7. Test Creation Speed | Requires typing out every line of code, using complex locator syntax. | Very fast (GUI/Autopilot). | Very fast (NLP/Recorder). |
| 8. Handling Complex Logic | Highly Flexible. Code can handle any complex logic or custom functions. | Limited. May lead to "many nested branches", making large tests difficult to manage. | Moderate to Good. Supports custom functions and scripting for advanced logic, bridging the gap between Low-Code and pure Code. |
| 9. Mobile Testing | Good (Web, API, Native Apps - requires config). | Very Good (Web, iOS, Android). | Very Good (Web, iOS, Android). |

Example test case:

- Login to <https://www.saucedemo.com/>
- Add a product to cart

demo.spec.ts

```

● ● ● demo.spec.ts
  1 import { test, expect } from '@playwright/test';
  2
  3 test('login and add product to cart', async ({ page }) => {
  4   await page.goto('https://www.saucedemo.com/');
  5
  6   await page.getByPlaceholder("Username").fill("standard_user");
  7   await page.getByPlaceholder("Password").fill("secret_sauce");
  8
  9   await page.getByRole("button", { name: "Login" }).click();
 10
 11  await expect(page).toHaveURL(/.*inventory.html/);
 12
 13  await expect(page.getByText("Products")).toBeVisible();
 14
 15  await page
 16    .locator(".inventory_item")
 17    .filter({ hasText: "Sauce Labs Backpack" })
 18    .getByRole("button", { name: "Add to cart" })
 19    .click();
 20
 21  await expect(page.locator(".shopping_cart_badge")).toHaveText("1");
 22 });
 23

```

[URL Demo MagicPod](#)
Process:

- Testers interact with the GUI, selecting predefined keywords/actions (e.g., Click, Input Text, Verify).
- Each selected keyword is then mapped directly to a specific underlying code function.

[URL Demo Testsigma](#)
Process:

- Testers directly type commands in simple English (e.g., Type "value" in the text field "Name")
- The NLP engine then analyzes the semantics (meaning) of the entire typed sentence and translates the intent into the executable action code.