

# Playwright TypeScript Framework

```
PLAYWRIGHT TYPESCRIPT FRAMEWORK (UIAction + POM)

==== UIAction CLASS (Recommended Locators) ====

import { Page } from "@playwright/test";

export class UIAction {
    constructor(protected page: Page) {}

    getByRole(role: string, name?: string) {
        return this.page.getByRole(role, name ? { name } : {});
    }

    getByLabel(label: string) {
        return this.page.getByLabel(label);
    }

    getByPlaceholder(text: string) {
        return this.page.getByPlaceholder(text);
    }

    getByText(text: string) {
        return this.page.getText(text);
    }

    getByTestId(id: string) {
        return this.page.getByTestId(id);
    }

    getByCss(selector: string) {
        return this.page.locator(selector);
    }

    getByXpath(xpath: string) {
        return this.page.locator(`xpath=${xpath}`);
    }

    getFrameByName(name: string) {
        return this.page.frameLocator(`iframe[name='${name}']");
    }
}

==== LOGIN PAGE (POM) ====

import { Page, expect } from "@playwright/test";
import { UIAction } from "./base/uiAction";

export class LoginPage extends UIAction {
    constructor(page: Page) {
        super(page);
    }

    async navigate() {
        await this.page.goto("https://example.com/login");
    }

    async enterUsername(username: string) {
        await this.getLabel("Username").fill(username);
    }
}
```

```

}

async enterPassword(password: string) {
  await this.getByLabel("Password").fill(password);
}

async clickLoginButton() {
  await this.getRole("button", "Login").click();
}

async verifyLoginSuccess() {
  await expect(this.getText("Welcome")).toBeVisible();
}
}

==== TEST FILE EXAMPLE ====

import { test } from "@playwright/test";
import { LoginPage } from "../pages/login.page";
import { testData } from "../utils/testData";

test("Verify user login functionality", async ({ page }) => {
  const loginPage = new LoginPage(page);

  await loginPage.navigate();
  await loginPage.enterUsername(testData.username);
  await loginPage.enterPassword(testData.password);
  await loginPage.clickLoginButton();
  await loginPage.verifyLoginSuccess();
});

==== TEST DATA ===

export const testData = {
  username: "testUser",
  password: "Test@123",
};

==== PLAYWRIGHT CONFIG ===

import { defineConfig } from '@playwright/test';

export default defineConfig({
  testDir: './tests',
  timeout: 60000,
  use: {
    headless: false,
    screenshot: "only-on-failure",
    video: "retain-on-failure",
  },
});

```