# Company House Automation Testing Project – User Instructions

# Project Overview

This automation project tests the website <https://automationintesting.online> using Playwright, TypeScript, and Cucumber BDD. It includes both functional and non-functional testing, covering:

|  |  |
| --- | --- |
| Type | Areas Covered |
| Functional Testing | Booking, Admin Panel, Navigation, Contact Form |
| API Testing | Booking, Authentication, Room Management, Messaging |
| Non-Functional Testing | Accessibility (WCAG 2.1), Performance, Mobile Support |

## Prerequisites

|  |  |
| --- | --- |
| Requirement | Description |
| Node.js | Version 18 or higher |
| npm | Package manager |
| Git | For cloning the repository |

## Project Setup

|  |  |
| --- | --- |
| Step | Command / Action |
| 1 | git clone https://github.com/your-username/CompanyHouse-ChrisAnajemba.git |
| 2 | cd CompanyHouse-ChrisAnajemba |
| 3 | npm install |
| 4 | npm run install-browsers |

## Running Tests

### Run All Tests

|  |  |
| --- | --- |
| Command | Purpose |
| npm test | Runs all BDD tests |
| npm run test:headed | Runs tests in a visible browser |
| npm run test | Launches the Playwright Test |

### Run Tests by Category

|  |  |
| --- | --- |
| Command | Test Scope |
| npm run test:booking @booking | Booking functionality |
| npm run test:admin @admin | Admin panel validation |
| npm run test:api @api | API endpoint checks |
| npm run test:accessibility @accessibility | WCAG 2.1 accessibility rules |
| npm run test:performance @performance | Load and response time |
| npm run test:contact @contact | Contact form validations |
| npm run test:navigation @navigation | Menu and section navigation |

### Run Individual Playwright Test Files

|  |  |
| --- | --- |
| Command | Test File |
| npx playwright test tests/booking-flow.spec.ts | Booking test |
| npx playwright test tests/accessibility.spec.ts | Accessibility checks |
| npx playwright test tests/api-testing.spec.ts | API testing suite |

### Smoke and Regression Suites

|  |  |
| --- | --- |
| Command | Suite Type |
| npm run test:smoke | Critical path tests |
| npm run test:regression | Full regression |

## Test Structure and Key Files

|  |  |
| --- | --- |
| Path | Description |
| features/ | Gherkin-based feature specs |
| features/step-definitions/ | Step implementation files |
| pages/ | Page Object Model (POM) classes |
| tests/ | Playwright spec files |
| reports/ | Generated HTML reports |
| screenshots/ | Screenshots for failed tests |
| test-results/ | Raw test outputs |
| playwright-report/ | Playwright HTML test report |
| docs/ | Documentation (test plan, bug reports, etc.) |
| playwright.config.ts | Playwright configuration |
| cucumber.js | Cucumber BDD configuration |
| package.json | Project dependencies and scripts |

## Notable Feature Files

|  |  |
| --- | --- |
| File Name | Purpose |
| booking.feature | Hotel booking workflow |
| accessibility.feature | WCAG accessibility validation |
| admin-access.feature | Admin login and security |
| contact-form.feature | Contact form behaviour |
| navigation.feature | Navigation link tests |
| performance.feature | Page and element performance |
| api-testing.feature | RESTful API validation |

## Key Page Object Files

|  |  |
| --- | --- |
| File Name | Purpose |
| BookingPage.ts | Booking form interactions |
| AdminLoginPage.ts | Admin login actions |
| ContactPage.ts | Contact form behavior |
| MenuPage.ts | Navigation and scrolling |
| AccessibilityHelper.ts | WCAG 2.1 utilities |

## Viewing Reports

### HTML Reports

|  |  |
| --- | --- |
| Command | Report Type |
| npm run view-reports | Opens all reports |
| npm run view-accessibility | Opens accessibility report |
| npm run view-performance | Opens performance summary |

### Playwright HTML Report

|  |  |
| --- | --- |
| Command | Description |
| npm run report | Opens latest test summary |
| npx playwright show-report | Alternative command |

### Screenshots

|  |  |
| --- | --- |
| Location | Description |
| /screenshots/ | Contains failure screenshots automatically |

## Advanced Usage

|  |  |
| --- | --- |
| Task | Command Example |
| Run in Firefox | npx playwright test --project=firefox |
| Run in WebKit (Safari) | npx playwright test --project=webkit |
| Debug Mode | npx playwright test --debug |
| Run tests with @tags | npm run test:booking |
| Generate All Reports | npm run generate-reports |
| Accessibility Report Only | npm run accessibility-report |
| Performance Report Only | npm run performance-report |

## Support & Documentation

|  |  |
| --- | --- |
| Resource | Location |
| Bug Reports | docs/Bug Reports.docx |
| Test Plans | docs/Test Plan.docx |
| Troubleshooting Info | playwright-report/, screenshots/ |

This framework enables complete test coverage of functional and non-functional requirements with modular architecture, parallel execution, rich reporting, and scalable test practices using Playwright, Cucumber BDD, and TypeScript.

Note: I have ignored the following files before committing this to GitHub, hence following this instruction from the setup will help to get the project ready for automation.

*Main reports are contained within the folder called ‘reports/ ‘, hence, ‘playwright-report/’ folder only works when not running the test on BDD. It is important to note that the framework is designed to run as a BDD (better test understanding by non-technical people and stakeholders) and as TDD (support sharding – though not implemented but great for pipeline execution as it saves time for quicker decision making).*

*# Playwright*

node\_modules/

/test-results/

/playwright-report/

/blob-report/

/playwright/.cache/