

Pa11y Configuration Guide: JSON Options and Testing Scenarios

Pa11y is a powerful accessibility testing tool that offers extensive JSON configuration capabilities for diverse testing scenarios. This guide provides comprehensive coverage of configuration options, real-world examples, and implementation patterns for various use cases.

JSON Configuration Structure

Pa11y supports two primary configuration formats: **pa11y.json** for command-line usage and **.pa11yrci** for CI/CD environments. Both use identical JSON schema with flexible option inheritance.

Basic Configuration Format

```
{
  "defaults": {
    "timeout": 30000,
    "standard": "WCAG2AA",
    "viewport": {
      "width": 1280,
      "height": 1024
    },
    "ignore": ["notice", "warning"]
  },
  "urls": [
    "https://example.com",
    {
      "url": "https://example.com/special",
      "timeout": 60000,
      "actions": ["click element #menu-toggle"]
    }
  ]
}
```

Complete Configuration Options Reference

Core Testing Options

standard (String): Accessibility standard to test against - Values: "WCAG2A", "WCAG2AA", "WCAG2AAA", "Section508" - Default: "WCAG2AA" - Note: Section 508 is deprecated; use WCAG2AA instead

runners (Array): Testing engines to use - Values: ["htmlcs"], ["axe"], ["htmlcs", "axe"] - Default: ["htmlcs"] - Multiple runners provide comprehensive coverage

actions (Array): Pre-test interactive actions - Format: Array of action strings - Available commands: `click element`, `set field`, `wait for element`, `navigate to` - Essential for testing authenticated or dynamic content

Browser and Performance Controls

timeout (Number): Maximum test duration in milliseconds - Default: 30000 (30 seconds) - Range: 1000 to 500000 for complex applications

viewport (Object): Browser window configuration - Properties: `width`, `height`, `deviceScaleFactor`, `isMobile` - Default: `{"width": 1280, "height": 1024}`

chromeLaunchConfig (Object): Chrome browser launch options - Critical for Docker/CI environments - Common args: `["--no-sandbox", "--disable-dev-shm-usage"]`

Issue Management

ignore (Array): Rules or issue types to exclude - Format: Specific rule codes or general types - Examples: `["WCAG2AA.Principle1.Guideline1_1.1_1_1.H30.2", "warning"]`

threshold (Number): Maximum allowed issues before failure - Default: 0 (strict mode) - Useful for gradual accessibility improvements

level (String): Minimum issue severity to report - Values: `"error"`, `"warning"`, `"notice"` - Default: `"error"`

Authentication and Headers

headers (Object): HTTP headers for requests - Format: Key-value pairs - Common use: `{"Authorization": "Bearer token", "Cookie": "session=abc123"}`

userAgent (String): Custom User-Agent string - Default: `"p11y/<version>"`

Accessibility Standards Configuration

WCAG Compliance Levels

```
{
  "defaults": {
    "standard": "WCAG2AA",
    "runners": ["htmlcs", "axe"],
    "level": "error",
    "includeWarnings": true,
    "includeNotices": true
  }
}
```

Custom WCAG 2.1 Rules

```
{
  "defaults": {
    "standard": "WCAG2A",
    "rules": [
      "Principle1.Guideline1_4.1_4_6_AAA",
      "Principle1.Guideline1_3.1_3_1_AAA"
    ]
  }
}
```

Government/Enterprise Compliance

```
{
  "defaults": {
    "standard": "WCAG2AA",
    "runners": ["htmlcs", "axe"],
    "threshold": 0,
    "level": "warning",
    "ignore": [
      "WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail"
    ]
  }
}
```

Authentication Scenarios

Basic Authentication

```
{
  "defaults": {
    "headers": {
      "Authorization": "Basic dXNlcm5hbWU6cGFzc3dvcmQ="
    }
  }
}
```

Form-Based Login

```
{
  "defaults": {
    "timeout": 60000,
    "useIncognitoBrowserContext": false
  },
  "urls": [
    {

```

```

        "url": "https://example.com/dashboard",
        "actions": [
            "navigate to https://example.com/login",
            "set field #username to testuser",
            "set field #password to testpass",
            "click element #login-button",
            "wait for path to be /dashboard"
        ]
    }
]
}

```

Multi-Step Authentication

```

{
  "urls": [
    {
      "url": "https://example.com/secure-area",
      "actions": [
        "navigate to https://example.com/login",
        "set field #username to admin",
        "set field #password to password123",
        "click element #login-btn",
        "wait for element #two-factor-input to be visible",
        "set field #two-factor-input to 123456",
        "click element #verify-btn",
        "wait for path to be /dashboard"
      ]
    }
  ]
}

```

Viewport and Device Testing

Mobile-First Testing

```

{
  "defaults": {
    "viewport": {
      "width": 375,
      "height": 667,
      "deviceScaleFactor": 2,
      "isMobile": true
    }
  }
}

```

Multi-Device Testing

```
{
  "urls": [
    {
      "url": "https://example.com",
      "viewport": {"width": 1920, "height": 1080}
    },
    {
      "url": "https://example.com",
      "viewport": {"width": 768, "height": 1024}
    },
    {
      "url": "https://example.com",
      "viewport": {"width": 375, "height": 667}
    }
  ]
}
```

Output Formats and Reporting

Multiple Reporter Configuration

```
{
  "defaults": {
    "reporters": [
      "cli",
      "json",
      ["pally-ci-reporter-html", {"fileName": "./report.html"}]
    ]
  }
}
```

Custom Output Paths

```
{
  "defaults": {
    "screenCapture": "screenshots/accessibility-test.png",
    "reporter": "json"
  }
}
```

CI/CD Integration Patterns

Jenkins Configuration

```
{
  "defaults": {
```

```

    "chromeLaunchConfig": {
      "args": [
        "--no-sandbox",
        "--disable-setuid-sandbox",
        "--disable-dev-shm-usage"
      ]
    },
    "timeout": 30000,
    "threshold": 0,
    "level": "error"
  }
}

```

GitHub Actions Optimized

```

{
  "defaults": {
    "timeout": 30000,
    "threshold": 0,
    "level": "error",
    "concurrency": 2,
    "chromeLaunchConfig": {
      "args": ["--no-sandbox", "--disable-setuid-sandbox"]
    }
  }
}

```

Docker Environment

```

{
  "defaults": {
    "chromeLaunchConfig": {
      "executablePath": "/usr/bin/google-chrome-stable",
      "args": [
        "--no-sandbox",
        "--disable-dev-shm-usage",
        "--disable-gpu",
        "--disable-extensions"
      ]
    }
  }
}

```

Performance and Timing Configurations

High-Performance Setup

```
{
  "defaults": {
    "timeout": 500000,
    "wait": 2000,
    "concurrency": 4,
    "useIncognitoBrowserContext": true,
    "chromeLaunchConfig": {
      "args": [
        "--disable-background-timer-throttling",
        "--disable-backgrounding-occluded-windows",
        "--disable-renderer-backgrounding"
      ]
    }
  }
}
```

Resource-Constrained Environment

```
{
  "defaults": {
    "timeout": 15000,
    "concurrency": 1,
    "chromeLaunchConfig": {
      "args": ["--disable-gpu", "--disable-extensions"]
    }
  }
}
```

Multi-Page Testing Configurations

Site-Wide Testing

```
{
  "defaults": {
    "timeout": 30000,
    "standard": "WCAG2AA",
    "hideElements": "#cookie-banner, .ad-container",
    "ignore": ["notice", "warning"]
  },
  "urls": [
    "https://example.com/",
    "https://example.com/about",
    "https://example.com/contact",
  ]
}
```

```

    {
      "url": "https://example.com/checkout",
      "timeout": 60000,
      "actions": [
        "navigate to https://example.com/login",
        "set field #email to test@example.com",
        "set field #password to testpass",
        "click element #login-btn"
      ]
    }
  ]
}

```

Sitemap-Based Testing

Command usage with configuration

```
pa11y-ci --sitemap https://example.com/sitemap.xml --config .pa11y-ci
```

```

{
  "defaults": {
    "timeout": 30000,
    "threshold": 5,
    "ignore": ["notice", "warning"]
  }
}

```

Custom Rules and Advanced Filtering

Comprehensive Issue Management

```

{
  "defaults": {
    "standard": "WCAG2AA",
    "ignore": [
      "WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail",
      "WCAG2AA.Principle1.Guideline1_4.1_4_3.G145.Fail",
      "WCAG2AA.Principle3.Guideline3_2.3_2_2.H32.2"
    ],
    "hideElements": "#lc_chat_layout, .third-party-widget",
    "threshold": 5,
    "includeWarnings": true
  }
}

```


Per-URL Custom Rules

```
{
  "urls": [
    {
      "url": "https://example.com/article",
      "hideElements": "#carbonads, #disqus_thread",
      "ignore": ["WCAG2AA.Principle1.Guideline1_4.1_4_3.G18.Fail"]
    }
  ]
}
```

Environment-Specific Configurations

Development Environment

```
// .pallyci.js for dynamic configuration
module.exports = {
  defaults: {
    timeout: process.env.PA11Y_TIMEOUT || 30000,
    headers: {
      'Authorization': `Bearer ${process.env.AUTH_TOKEN}`
    },
    chromeLaunchConfig: {
      args: process.env.CI ? ['--no-sandbox'] : []
    }
  },
  urls: [
    `${process.env.BASE_URL || 'https://localhost:3000'}/`
  ]
};
```

Production Monitoring

```
{
  "defaults": {
    "timeout": 45000,
    "threshold": 0,
    "ignore": ["notice"],
    "screenCapture": "screenshots/prod-test.png",
    "headers": {
      "User-Agent": "Pa1ly-Monitor/1.0"
    }
  }
}
```

Advanced Action Sequences

Complex User Flows

```
{
  "urls": [
    {
      "url": "https://example.com/multi-step-form",
      "actions": [
        "set field #step1-name to John Doe",
        "click element #next-step",
        "wait for element #step2-form to be visible",
        "set field #step2-email to john@example.com",
        "click element #submit-form",
        "wait for path to be /confirmation",
        "screen capture form-completion.png"
      ]
    }
  ]
}
```

Dynamic Content Testing

```
{
  "urls": [
    {
      "url": "https://example.com/dynamic-content",
      "actions": [
        "click element #load-more-button",
        "wait for element .dynamic-content to be visible",
        "click element #filter-toggle",
        "wait for element #filter-options to be visible"
      ]
    }
  ]
}
```

Integration with Testing Frameworks

Jest Integration

```
// jest.config.js
module.exports = {
  testEnvironment: 'node',
  setupFilesAfterEnv: ['<rootDir>/test/pa11y.setup.js']
};
```

```
// pally.setup.js
const pally = require('pally');
const config = require('./pally.json');

global.pallyTest = async (url, options = {}) => {
  const results = await pally(url, { ...config.defaults, ...options });
  return results;
};
```

Best Practices for Configuration

Configuration organization ensures maintainability and clarity across different testing scenarios. Environment separation uses different configuration files for development, staging, and production environments. Modular configuration breaks complex setups into reusable components with consistent naming conventions.

Security considerations require protecting authentication credentials through environment variables and avoiding hardcoded sensitive data in configuration files. **Performance optimization** balances thoroughness with execution time by using appropriate concurrency levels and timeout values.

Maintenance strategies include regular review of ignore rules to ensure they remain relevant, documentation of custom configurations for team knowledge sharing, and version control of configuration files alongside application code.

This comprehensive configuration guide demonstrates pally's extensive flexibility for accessibility testing across diverse scenarios, from basic compliance checking to complex enterprise deployments with authentication, custom rules, and CI/CD integration.