

Automation Testing CI/CD Guide PDF Page

Automation Testing CI/CD Guide PDF Page: Content + <head> (Uncodemy)

FINAL URL: <https://uncodemy.com/automation-testing-ci-cd-guide-pdf>

Publish this page at exactly this URL (non-www, no trailing slash). Canonical, og:url and sitemap entry must match it.

Prepared: 26 October 2026 · **Primary keyword:** automation testing ci cd guide pdf · **Search volume:** not checked (keyword tool out of credits); verify in Search Console.

Read first (5 flags):

- Your course PDF title, page count and size were not shared.** Download cards are fill-in slots ([CONFIRM]). Card 1 is your course PDF; cards 2 and 3 only if you have more files.
- One unique file per button.** The same course PDF repeated on many pages is fine as a resource, but each page's text must stay unique and useful without the file.
- Free or gated?** Not stated. The page says "Download", not "Free". If a form is used, add a consent checkbox and Privacy Policy link (DPDP Act 2023).
- No made-up stats or testimonials;** no placement or salary claims without a named, dated source.
- Author and reviewer are placeholders;** use real named trainers and one real "Last reviewed" date.

PART 1: Page plan

Item	Decision
Final URL	https://uncodemy.com/automation-testing-ci-cd-guide-pdf
Intent (anti-cannibalisation)	RUNNING AUTOMATED TESTS IN CI/CD. Framework-specific pages (TestNG, Cucumber, POM) are separate.
Why not thin	Explainer-style sections, FAQs and cross-links make the page useful even without a download
Schema	WebSite, Organization, Breadcrumb, CollectionPage, FAQPage (only FAQs visible on the page). PDF ItemList provided commented, to enable after titles are final
Share image	https://uncodemy.com/image/logo-og.jpg for now; custom 1200x630 card recommended

PART 2: Page content (copy-paste ready)

SEO Title (45 chars):

Automation Testing CI/CD Guide PDF | Uncodemy

Meta Description (131 chars):

Download the automation testing CI/CD guide PDF from Uncodemy: pipeline stages, a sample workflow, reports and flaky test handling.

H1: Automation Testing CI/CD Guide PDF Downloads **Breadcrumb:** Home > Automation Testing CI/CD Guide PDF **Final URL:** <https://uncodemy.com/automation-testing-ci-cd-guide-pdf>

Byline: Prepared by the Uncodemy trainer team [CONFIRM: author name and role] · Reviewed by [CONFIRM: reviewer name and role] · Last reviewed: [DATE]

Quick Answer

This page offers Uncodemy's automation testing CI/CD guide as a PDF download. Continuous integration runs your automated tests every time code changes, giving fast feedback. The guide explains pipeline stages, which tests to run where, a sample workflow, how to publish reports and how to deal with slow and flaky tests.

Download Automation Testing CI/CD Guide PDF

[CONFIRM: free or gated, and whether details are needed before download]

PDF 1 (main): [CONFIRM TITLE: your Software Testing course PDF]
[CONFIRM: one line on what it covers and who it is for] · [N] pages · [X] MB · PDF · Updated [DATE]
[Download PDF] → <https://uncodemy.com/pdf/automation-testing-ci-cd-guide.pdf> (your course PDF, uploaded under this file name)

PDF 2 (optional): [CONFIRM TITLE]
Only if you have a second PDF that genuinely fits this topic. **[Download PDF]** → <https://uncodemy.com/pdf/automation-testing-ci-cd-guide-2.pdf>

PDF 3 (optional): [CONFIRM TITLE]
Same rule. **[Download PDF]** → <https://uncodemy.com/pdf/automation-testing-ci-cd-guide-3.pdf>

Each button must open its own file. Do not point several buttons at one file. Delete unused optional cards.

Why tests belong in a pipeline

Automated tests only deliver value when they run. A suite that someone runs by hand once a week finds problems late, when the change that caused them is hard to trace. Continuous integration, or CI, builds and tests the software automatically on every commit or pull request, so a failure points to a small, recent change. Continuous delivery extends this by keeping the software always in a releasable state and moving it through environments automatically. For testers this means automation is not a side project: tests become part of the delivery process, with the same expectations of reliability, speed and ownership as application code.

Pipeline stages and the tests that fit each

Stage	Typical tests	Aim
Commit or pull request	Unit tests, linting, static analysis	Seconds to a few minutes of feedback
Build and package	Compile, dependency checks	A deployable artifact
Deploy to test environment	Smoke tests	Confirm the build is stable
Integration stage	API and integration tests	Verify services work together
Regression stage	UI end-to-end tests on key flows	Protect critical journeys
Nightly or pre-release	Full regression, cross-browser, performance and security checks	Deeper assurance without slowing every commit
Post-deploy	Smoke tests in the target environment, monitoring	Confirm the release works

Keep the fast feedback loop fast

Developers stop paying attention to a pipeline that takes an hour. Put the quickest, most reliable tests on every commit and move slower ones to later stages or schedules. Follow the test pyramid: many unit tests, a moderate number of API tests and a small set of end-to-end tests for essential user journeys. Run tests in parallel, share setup where it is safe and avoid unnecessary browser tests for logic that an API or unit test can cover. A target such as pull-request checks finishing within roughly ten minutes is a common aim, though the right number depends on your team.

Sample GitHub Actions workflow for a Maven test project

A minimal workflow that runs tests on every push and pull request and keeps the reports. Adapt versions, commands and paths to your project.

```
name: tests
on:
  push:
    branches: [ main ]
  pull_request:

jobs:
  test:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - uses: actions/setup-java@v4
        with:
          distribution: temurin
          java-version: '17'
          cache: maven
      - name: Run tests
        run: mvn -B test
      - name: Upload test reports
        if: always()
        uses: actions/upload-artifact@v4
        with:
          name: test-reports
          path: target/surefire-reports/
```

Setting up automation in CI step by step

1. Make the tests runnable from one command, with no manual setup or local paths.
2. Run browsers in headless mode in CI and pin the browser and driver versions.
3. Create a pipeline file in the repository so the process is versioned with the code.
4. Start with smoke tests on every build, then add regression stages gradually.
5. Publish reports and logs as artifacts, and send failures to the team chat or email.
6. Set rules: failing tests block the merge, and failures are fixed or quarantined promptly.

Test data and environments

Concern	Good practice
Environment stability	Use a dedicated, reset-able test environment; avoid sharing with manual experiments
Test data	Create data in the test itself through an API or fixtures, and clean up afterwards
Secrets	Store credentials in the CI tool's secret store, never in the repository
Third-party services	Use mocks or sandboxes to avoid flakiness and cost
Configuration	Pass the base URL, browser and environment as parameters

Handling flaky tests

A flaky test passes and fails on the same code. Flakiness destroys trust: people start re-running pipelines and ignoring red builds. Common causes are fixed sleeps instead of proper waits, tests that depend on one another, shared data, unstable environments and race conditions in the application itself. Investigate the cause rather than only adding retries. If a test cannot be fixed quickly, move it to a quarantine group that does not block merges, track it and fix or delete it. A small automatic retry can help with genuine infrastructure noise, but it should be visible in the report so that real problems are not hidden.

Reports and metrics worth publishing

Output	Why it helps
Pass and fail summary per run	Quick status for the team
Failure details with logs and screenshots	Faster diagnosis
Trend of pass rate and duration	Spot growing slowness or instability
Flaky test list	Prioritise cleanup
Coverage of critical flows	Show where protection is missing

Common mistakes

- **Running the whole UI suite on every commit**, making feedback slow.
- **Ignoring red builds** until they become normal.
- **Hard-coded environment details** that work only on one machine.
- **No ownership** for failing tests.
- **Treating retries as a fix** for flakiness.

A simple adoption roadmap

- Month 1: run smoke tests on every build.
- Month 2: add API tests and a small end-to-end set for critical flows.
- Month 3: add nightly regression and flaky-test tracking.
- After that: review duration and failure trends monthly.

Related Uncodemy resources

- [Software Testing Regression Testing PDF](#)
- [Smoke and Sanity Testing PDF](#)
- [Automation Testing TestNG Guide PDF](#)
- [Playwright Page Object Model Guide PDF](#)

Learn with a trainer [use only the line that is true]

- **Software Testing course:** [CONFIRM URL]. [CONFIRM URL] Link only if this course page is live; otherwise use <https://uncodemy.com/all-courses>

(One soft call to action: course link or free demo. No pop-up.)

About these PDFs

These PDFs are study material for learning and revision. They are not a certificate, a degree or a promise of a job, and technology changes, so check official documentation for the latest details. Spot an error? Write to [CONFIRM: feedback email] and the trainer team will review it.

Frequently Asked Questions

What is CI/CD in test automation?

It means running automated tests automatically in a pipeline whenever code changes, and keeping software ready for release through automated build, test and deployment stages.

Which tests should run on every commit?

Fast, reliable tests such as unit tests, static checks and a small smoke suite. Slower end-to-end and cross-browser suites usually run later or on a schedule.

How do I run Selenium or Playwright tests in CI?

Run the browser in headless mode, pin versions, use a single command to start the tests and upload reports and screenshots as pipeline artifacts.

What is a flaky test and how do I fix it?

A flaky test gives inconsistent results on the same code. Fix the cause, such as improper waits or shared data, quarantine it if needed and avoid hiding it with endless retries.

Which CI tools are commonly used for test automation?

Common choices include Jenkins, GitHub Actions, GitLab CI and Azure DevOps. The concepts are the same across them.

Are these PDFs free to download?

[CONFIRM: state clearly whether the PDFs are free, or whether a short form is needed first.]

The last FAQ contains [CONFIRM], so it is NOT in the FAQ schema. Fill it and keep it on the page, or delete it.

PART 3: <head> section (copy-paste ready)

Final URL: <https://uncodemy.com/automation-testing-ci-cd-guide-pdf>

```
<!-- =====
UNCODEMY <head> | FINAL URL: https://uncodemy.com/automation-testing-ci-cd-guide-pdf
Prepared 26 October 2026. Paste into the page <head>; keep your site-wide favicon, fonts, analytics.
BEFORE GO-LIVE: (1) set dateModified in the JSON-LD to the real publish date;
(2) replace og:image with a 1200x630 share card when available;
(3) once PDF titles are final, uncomment the ItemList block at the bottom.
===== -->

<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Automation Testing CI/CD Guide PDF | Uncodemy</title>
<meta name="description" content="Download the automation testing CI/CD guide PDF from Uncodemy: pipeline stages, a sample workflow, reports and flaky test
handling.">
<link rel="canonical" href="https://uncodemy.com/automation-testing-ci-cd-guide-pdf">
<meta name="robots" content="index, follow, max-snippet:-1, max-image-preview:large, max-video-preview:-1">
<meta name="author" content="Uncodemy">
<meta name="publisher" content="Uncodemy Edutech Pvt. Ltd.">
<meta name="language" content="en-IN">
<link rel="alternate" hreflang="en-in" href="https://uncodemy.com/automation-testing-ci-cd-guide-pdf">
<link rel="alternate" hreflang="x-default" href="https://uncodemy.com/automation-testing-ci-cd-guide-pdf">

<!-- Open Graph -->
<meta property="og:type" content="website">
<meta property="og:site_name" content="Uncodemy">
<meta property="og:locale" content="en_IN">
<meta property="og:url" content="https://uncodemy.com/automation-testing-ci-cd-guide-pdf">
<meta property="og:title" content="Automation Testing CI/CD Guide PDF | Uncodemy">
<meta property="og:description" content="Download the automation testing CI/CD guide PDF from Uncodemy: pipeline stages, a sample workflow, reports and flaky
handling.">
<meta property="og:image" content="https://uncodemy.com/image/logo-og.jpg">
<meta property="og:image:alt" content="Uncodemy Automation Testing CI/CD Guide PDF study material">

<!-- X (Twitter) -->
<meta name="twitter:card" content="summary_large_image">
<meta name="twitter:site" content="@uncodemy">
<meta name="twitter:title" content="Automation Testing CI/CD Guide PDF | Uncodemy">
<meta name="twitter:description" content="Download the automation testing CI/CD guide PDF from Uncodemy: pipeline stages, a sample workflow, reports and flaky
test handling.">
<meta name="twitter:image" content="https://uncodemy.com/image/logo-og.jpg">
<meta name="twitter:image:alt" content="Uncodemy Automation Testing CI/CD Guide PDF study material">

<!-- Structured data: one @graph (WebSite, Organization, Breadcrumb, CollectionPage, FAQPage) -->
<script type="application/ld+json">
{
  "@context": "https://schema.org",
  "@graph": [
    {
      "@type": "WebSite",
      "@id": "https://uncodemy.com/#website",
      "url": "https://uncodemy.com/",
      "name": "Uncodemy",
      "publisher": {
        "@id": "https://uncodemy.com/#organization"
      },
      "inLanguage": "en-IN"
    },
    {
      "@type": "EducationalOrganization",
      "@id": "https://uncodemy.com/#organization",
      "name": "Uncodemy",
      "legalName": "Uncodemy Edutech Pvt. Ltd.",
      "url": "https://uncodemy.com/",
      "logo": {
        "@type": "ImageObject",
        "url": "https://uncodemy.com/uc-layout/img/Logo.webp"
      },
      "email": "info@uncodemy.com",
      "identifier": {
        "@type": "PropertyValue",
        "name": "CIN",
        "value": "U85499UP2024PTC195902"
      },
      "contactPoint": [
        {
          "@type": "ContactPoint",
          "telephone": "+91-9818366550",
          "contactType": "admissions",
          "areaServed": "IN",
          "availableLanguage": [
            "en",
            "hi"
          ]
        }
      ],
      "sameAs": [
        "https://x.com/uncodemy",
        "https://www.facebook.com/uncodemyofficial/",
        "https://www.linkedin.com/company/uncodemy/",
        "https://www.instagram.com/uncodemyofficial/"
      ]
    },
    {
      "@type": "BreadcrumbList",
      "@id": "https://uncodemy.com/automation-testing-ci-cd-guide-pdf#breadcrumb",
      "itemListElement": [
        {
          "@type": "ListItem",
          "position": 1,
          "name": "Home",
          "item": "https://uncodemy.com/"
        },
        {
          "@type": "ListItem",
          "position": 2,
          "name": "Automation Testing CI/CD Guide PDF",

```

```

        "item": "https://uncodemy.com/automation-testing-ci-cd-guide-pdf"
    }
},
{
    "@type": "CollectionPage",
    "@id": "https://uncodemy.com/automation-testing-ci-cd-guide-pdf#webpage",
    "url": "https://uncodemy.com/automation-testing-ci-cd-guide-pdf",
    "name": "Automation Testing CI/CD Guide PDF Downloads",
    "description": "Download the automation testing CI/CD guide PDF from Uncodemy: pipeline stages, a sample workflow, reports and flaky test handling.",
    "inLanguage": "en-IN",
    "isPartOf": {
        "@id": "https://uncodemy.com/#website"
    },
    "publisher": {
        "@id": "https://uncodemy.com/#organization"
    },
    "breadcrumb": {
        "@id": "https://uncodemy.com/automation-testing-ci-cd-guide-pdf#breadcrumb"
    },
    "about": {
        "@type": "Thing",
        "name": "Automation Testing CI/CD Guide"
    },
    "primaryImageOfPage": {
        "@type": "ImageObject",
        "url": "https://uncodemy.com/image/logo-og.jpg"
    },
    "dateModified": "2026-10-26"
},
{
    "@type": "FAQPage",
    "@id": "https://uncodemy.com/automation-testing-ci-cd-guide-pdf#faq",
    "inLanguage": "en-IN",
    "isPartOf": {
        "@id": "https://uncodemy.com/automation-testing-ci-cd-guide-pdf#webpage"
    },
    "mainEntity": [
        {
            "@type": "Question",
            "name": "What is CI/CD in test automation?",
            "acceptedAnswer": {
                "@type": "Answer",
                "text": "It means running automated tests automatically in a pipeline whenever code changes, and keeping software ready for release through automa
build, test and deployment stages."
            }
        },
        {
            "@type": "Question",
            "name": "Which tests should run on every commit?",
            "acceptedAnswer": {
                "@type": "Answer",
                "text": "Fast, reliable tests such as unit tests, static checks and a small smoke suite. Slower end-to-end and cross-browser suites usually run la
or on a schedule."
            }
        },
        {
            "@type": "Question",
            "name": "How do I run Selenium or Playwright tests in CI?",
            "acceptedAnswer": {
                "@type": "Answer",
                "text": "Run the browser in headless mode, pin versions, use a single command to start the tests and upload reports and screenshots as pipeline
artifacts."
            }
        },
        {
            "@type": "Question",
            "name": "What is a flaky test and how do I fix it?",
            "acceptedAnswer": {
                "@type": "Answer",
                "text": "A flaky test gives inconsistent results on the same code. Fix the cause, such as improper waits or shared data, quarantine it if needed a
avoid hiding it with endless retries."
            }
        },
        {
            "@type": "Question",
            "name": "Which CI tools are commonly used for test automation?",
            "acceptedAnswer": {
                "@type": "Answer",
                "text": "Common choices include Jenkins, GitHub Actions, GitLab CI and Azure DevOps. The concepts are the same across them."
            }
        }
    ]
}
]
}
</script>

<!-- ADD AFTER THE PDF TITLES ARE FINAL (remove this wrapper, fill every [...] value):
<script type="application/ld+json">
{
    "@context": "https://schema.org",
    "@type": "ItemList",
    "@id": "https://uncodemy.com/automation-testing-ci-cd-guide-pdf#pdfs",
    "name": "Automation Testing CI/CD Guide PDF downloads",
    "numberOfItems": 3,
    "itemListElement": [
        {
            "@type": "ListItem",
            "position": 1,
            "item": {
                "@type": "DigitalDocument",
                "name": "[PDF 1 TITLE]",
                "description": "[PDF 1 ONE-LINE DESCRIPTION]",
                "encodingFormat": "application/pdf",
                "url": "https://uncodemy.com/pdf/automation-testing-ci-cd-guide.pdf",
                "inLanguage": "en",
                "publisher": {
                    "@id": "https://uncodemy.com/#organization"
                }
            }
        }
    ]
}
},

```

```

    {
      "@type": "ListItem",
      "position": 2,
      "item": {
        "@type": "DigitalDocument",
        "name": "[PDF 2 TITLE]",
        "description": "[PDF 2 ONE-LINE DESCRIPTION]",
        "encodingFormat": "application/pdf",
        "url": "https://uncodemy.com/pdf/automation-testing-ci-cd-guide-2.pdf",
        "inLanguage": "en",
        "publisher": {
          "@id": "https://uncodemy.com/#organization"
        }
      }
    },
    {
      "@type": "ListItem",
      "position": 3,
      "item": {
        "@type": "DigitalDocument",
        "name": "[PDF 3 TITLE]",
        "description": "[PDF 3 ONE-LINE DESCRIPTION]",
        "encodingFormat": "application/pdf",
        "url": "https://uncodemy.com/pdf/automation-testing-ci-cd-guide-3.pdf",
        "inLanguage": "en",
        "publisher": {
          "@id": "https://uncodemy.com/#organization"
        }
      }
    }
  ]
}
</script>
-->

```

DEVELOPER CHECKLIST

- ☐ Final URL live at `https://uncodemy.com/automation-testing-ci-cd-guide-pdf`; canonical, og:url and sitemap identical.
- ☐ Each button: `<a href="/pdf/automation-testing-ci-cd-guide.pdf" download type="application/pdf">`, its own file, size shown.
- ☐ If a form precedes download: consent checkbox, Privacy Policy link, nothing pre-ticked.
- ☐ One `<h1>`; real HTML tables; code samples in `<pre><code>`.
- ☐ Remove any `<meta name="keywords">` from the template.
- ☐ Test in Google Rich Results Test and validator.schema.org after publishing.
- ☐ Add to sitemap and any PDF/resources hub.

STILL NEEDED FROM YOU

#	Item
1	Course PDF title, description, page count, size, and the file URL for each page
2	Free or gated (and form fields if gated)
3	Author, reviewer, publish and last-reviewed dates
4	Course URL for the CTA (lines marked <code>[CONFIRM]</code>)
5	Feedback email
6	Confirmation that Uncodemy owns the rights to every PDF