

Software Testing Bug Life Cycle PDF Page

Software Testing Bug Life Cycle PDF Page: Content + `<head>` (Uncodemy)

FINAL URL: <https://uncodemy.com/software-testing-bug-life-cycle-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:** software testing bug life cycle pdf · **Search volume:** not checked (keyword tool out of credits); verify in Search Console.

Read first (5 flags):

1. **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.
2. **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.
3. **Free or gated?** Not stated. The page says "Download", not "Free". If a form is used, add a consent checkbox and Privacy Policy link (DPPD Act 2023).
4. **No made-up stats or testimonials;** no placement or salary claims without a named, dated source.
5. **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/software-testing-bug-life-cycle-pdf
Intent (anti-cannibalisation)	DEFECT WORKFLOW and states. How to write the report itself is on the bug reporting page.
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 <code>ItemList</code> 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 (46 chars):

Software Testing Bug Life Cycle PDF | Uncodemy

Meta Description (137 chars):

Download the bug life cycle PDF from Uncodemy: defect states from New to Closed, special states, responsibilities and a workflow diagram.

H1: Software Testing Bug Life Cycle PDF Downloads **Breadcrumb:** Home > Software Testing Bug Life Cycle PDF **Final URL:** <https://uncodemy.com/software-testing-bug-life-cycle-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 bug life cycle guide as a PDF download. The bug life cycle, also called the defect life cycle, describes the states a defect passes through from the moment it is found until it is closed. The guide explains each state, who is responsible, the special states such as rejected, deferred and reopened, and how teams adapt the workflow.

Download Software Testing Bug Life Cycle 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/software-testing-bug-life-cycle.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/software-testing-bug-life-cycle-2.pdf>

PDF 3 (optional): [CONFIRM TITLE]
Same rule. **[Download PDF]** → <https://uncodemy.com/pdf/software-testing-bug-life-cycle-3.pdf>

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

What is the bug life cycle?

The bug life cycle is the set of states a defect moves through during its life in a tracking tool, together with the rules for moving between them. It exists so that everyone knows the status of every defect, who is responsible for the next action and what has been decided. Without a defined cycle, bugs are lost, fixed twice, closed without verification or argued about in chat. The exact names and steps vary between organisations and tools, but the pattern is common: a tester reports, a lead triages, a developer fixes, a tester verifies and the defect is closed.

Typical states and what they mean

State	Meaning	Usually moved by
New	A defect has been reported and is waiting for review	Tester
Assigned or Open	Accepted as valid and assigned to a developer	Test or development lead
In progress	The developer is working on a fix	Developer
Fixed or Resolved	The developer believes the defect is fixed and delivered in a build	Developer
Ready for retest	A build containing the fix is available for verification	Developer or release process
Verified or Closed	The tester confirmed the fix works and no regression appeared	Tester
Reopened	The fix did not work or the issue came back	Tester

Special states

State	When it is used	What should happen next
Rejected or Not a bug	Behaviour is as designed, or the report is invalid	Tester reviews the explanation and either accepts or discusses with the lead
Duplicate	The same defect is already reported	Link to the original and close this report
Deferred or Postponed	A valid defect that will be fixed in a later release	Record the target release; review at planning
Cannot reproduce	The developer cannot see the problem with the given steps	Tester adds details, environment and recordings, or retests
Blocked	The fix or retest depends on something else	Resolve the dependency; note the reason

A typical flow in words

A tester finds a defect, writes a report and saves it as New. The test lead reviews it, confirms it is valid and not a duplicate, sets severity and priority and assigns it to a developer, which moves it to Assigned. The developer investigates and, once working on it, marks it In progress. After the change is committed and built, the developer marks it Fixed and often adds a note about the cause and the change. The tester retests on the new build, repeating the original steps and checking nearby behaviour. If the problem is gone, the defect is Closed. If it remains, the tester marks it Reopened with a comment and the cycle repeats from assignment.

The workflow as text

A simple diagram showing the main path and the common side paths. Teams adapt the names to their tool.

New -> Assigned -> In progress -> Fixed -> Ready for retest -> Closed	
	+--> Reopened -> Assigned (cycle repeats)
New -> Rejected	(not a bug / as designed)
New -> Duplicate	(link to the original report)
Assigned -> Deferred	(fix planned for a later release)
In progress -> Cannot reproduce -> New	(tester adds detail)

Responsibilities by role

- Tester: reports clearly, retests promptly, reopens with evidence and closes only after verifying.
- Test or development lead: triages, sets priority, assigns and resolves disputes.
- Developer: reproduces, fixes, explains the cause and delivers the fix in a named build.
- Product owner: decides on priority and on whether to defer a defect.

Measures teams watch in the defect cycle

Measure	What it can reveal
Time in each state	Bottlenecks, such as defects waiting for retest
Reopen rate	Quality of fixes or of reproduction steps
Rejected rate	Clarity of requirements or tester training needs
Age of open defects	Backlog health
Defects by severity and module	Where quality problems cluster

Good practice

- Keep the workflow simple; too many states cause confusion.
- Require a comment whenever a state changes, especially for rejected, deferred and reopened.
- Always retest on the build that contains the fix and run a quick regression check around it.
- Review deferred defects regularly so they are not forgotten.

Common mistakes

- **Closing a defect without retesting.**
- **Rejecting without explanation**, which causes friction.
- **Leaving defects in Fixed for weeks** because nobody owns retesting.
- **Using the tracker as a chat tool** instead of keeping decisions in the record.
- **Different teams using the same state names with different meanings.**

Example: one defect through the cycle

Day	State	What happened
Mon	New	Tester reports that the coupon field accepts expired codes
Mon	Assigned	Lead confirms, sets severity High and assigns to a developer
Tue	In progress	Developer finds that the expiry date check uses the wrong time zone
Wed	Fixed	Fix merged and included in build 142
Thu	Reopened	Tester finds codes expiring at midnight still work in one region
Fri	Closed	Second fix verified on build 145 with regression checks

Related Uncodemy resources

- [Software Testing Bug Reporting Guide PDF](#)
- [Software Testing Regression Testing PDF](#)
- [Software Testing in Agile PDF](#)
- [Software Testing Test Plan 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 the bug life cycle?

It is the sequence of states a defect goes through from being reported to being closed, such as New, Assigned, Fixed, Retested and Closed.

What does Reopened mean in the bug life cycle?

Reopened means the tester found that the defect is not fixed or has come back after a fix, so it returns to the developer.

What is a deferred bug?

A valid defect whose fix is postponed to a later release, usually because of low priority or limited time.

Who closes a bug?

Usually the tester who verified the fix on a build that contains it, though some teams let the reporter or lead close it.

What is the difference between rejected and duplicate?

Rejected means the report is invalid or the behaviour is as designed. Duplicate means the defect is already reported elsewhere.

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/software-testing-bug-life-cycle-pdf>

```
<!-- =====
UNCODEMY <head> | FINAL URL: https://uncodemy.com/software-testing-bug-life-cycle-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>Software Testing Bug Life Cycle PDF | Uncodemy</title>
<meta name="description" content="Download the bug life cycle PDF from Uncodemy: defect states from New to Closed, special states, responsibilities and a work
diagram.">
<link rel="canonical" href="https://uncodemy.com/software-testing-bug-life-cycle-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/software-testing-bug-life-cycle-pdf">
<link rel="alternate" hreflang="x-default" href="https://uncodemy.com/software-testing-bug-life-cycle-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/software-testing-bug-life-cycle-pdf">
<meta property="og:title" content="Software Testing Bug Life Cycle PDF | Uncodemy">
<meta property="og:description" content="Download the bug life cycle PDF from Uncodemy: defect states from New to Closed, special states, responsibilities and
workflow diagram.">
<meta property="og:image" content="https://uncodemy.com/image/logo-og.jpg">
<meta property="og:image:alt" content="Uncodemy Software Testing Bug Life Cycle PDF study material">

<!-- X (Twitter) -->
<meta name="twitter:card" content="summary_large_image">
<meta name="twitter:site" content="@uncodemy">
<meta name="twitter:title" content="Software Testing Bug Life Cycle PDF | Uncodemy">
<meta name="twitter:description" content="Download the bug life cycle PDF from Uncodemy: defect states from New to Closed, special states, responsibilities an
workflow diagram.">
<meta name="twitter:image" content="https://uncodemy.com/image/logo-og.jpg">
<meta name="twitter:image:alt" content="Uncodemy Software Testing Bug Life Cycle 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/software-testing-bug-life-cycle-pdf#breadcrumb",
      "itemListElement": [
        {
          "@type": "ListItem",
          "position": 1,
          "name": "Home",
          "item": "https://uncodemy.com/"
        },
        {
          "@type": "ListItem",
          "position": 2,
          "name": "Software Testing Bug Life Cycle PDF",
          "item": "https://uncodemy.com/software-testing-bug-life-cycle-pdf"
        }
      ]
    },
    {
      "@type": "CollectionPage",
      "@id": "https://uncodemy.com/software-testing-bug-life-cycle-pdf#webpage",
      "url": "https://uncodemy.com/software-testing-bug-life-cycle-pdf",
      "name": "Software Testing Bug Life Cycle PDF Downloads",
      "description": "Download the bug life cycle PDF from Uncodemy: defect states from New to Closed, special states, responsibilities and a workflow diagram",
      "inLanguage": "en-IN",
      "isPartOf": {
        "@id": "https://uncodemy.com/#website"
      }
    }
  ],
}

```

```

    "publisher": {
      "@id": "https://uncodemy.com/#organization"
    },
    "breadcrumb": {
      "@id": "https://uncodemy.com/software-testing-bug-life-cycle-pdf#breadcrumb"
    },
    "about": {
      "@type": "Thing",
      "name": "Software Testing Bug Life Cycle"
    },
    "primaryImageOfPage": {
      "@type": "ImageObject",
      "url": "https://uncodemy.com/image/logo-og.jpg"
    },
    "dateModified": "2026-10-26"
  },
  {
    "@type": "FAQPage",
    "@id": "https://uncodemy.com/software-testing-bug-life-cycle-pdf#faq",
    "inLanguage": "en-IN",
    "isPartOf": {
      "@id": "https://uncodemy.com/software-testing-bug-life-cycle-pdf#webpage"
    },
    "mainEntity": [
      {
        "@type": "Question",
        "name": "What is the bug life cycle?",
        "acceptedAnswer": {
          "@type": "Answer",
          "text": "It is the sequence of states a defect goes through from being reported to being closed, such as New, Assigned, Fixed, Retested and Closed"
        }
      },
      {
        "@type": "Question",
        "name": "What does Reopened mean in the bug life cycle?",
        "acceptedAnswer": {
          "@type": "Answer",
          "text": "Reopened means the tester found that the defect is not fixed or has come back after a fix, so it returns to the developer."
        }
      },
      {
        "@type": "Question",
        "name": "What is a deferred bug?",
        "acceptedAnswer": {
          "@type": "Answer",
          "text": "A valid defect whose fix is postponed to a later release, usually because of low priority or limited time."
        }
      },
      {
        "@type": "Question",
        "name": "Who closes a bug?",
        "acceptedAnswer": {
          "@type": "Answer",
          "text": "Usually the tester who verified the fix on a build that contains it, though some teams let the reporter or lead close it."
        }
      },
      {
        "@type": "Question",
        "name": "What is the difference between rejected and duplicate?",
        "acceptedAnswer": {
          "@type": "Answer",
          "text": "Rejected means the report is invalid or the behaviour is as designed. Duplicate means the defect is already reported elsewhere."
        }
      }
    ]
  }
]
}
</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/software-testing-bug-life-cycle-pdf#pdfs",
  "name": "Software Testing Bug Life Cycle 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/software-testing-bug-life-cycle.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/software-testing-bug-life-cycle-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/software-testing-bug-life-cycle-3.pdf",
        "inLanguage": "en",
        "publisher": {
          "@id": "https://uncodemy.com/#organization"
        }
      }
    }
  }
}
</script>
-->
```

DEVELOPER CHECKLIST

- ☐ Final URL live at `https://uncodemy.com/software-testing-bug-life-cycle-pdf`; canonical, og:url and sitemap identical.
- ☐ Each button: `<a href="/pdf/software-testing-bug-life-cycle.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