BugIt
by Taskivator

QA Bug-Filing Agent for VS Code

BugIt

Turn rough test notes into polished, reviewed bug tickets, filed to your tracker in seconds.

A QA agent that learns your team's workflow.

Built by a senior QA engineer who writes and reviews bug reports every day.
Private by design: your work is never sent to Taskivator.

~10 min	Dupe check	~15 min	$59.99
saved per bug	before you file	one-time setup	Solo · 1 year · no subscription

Getting Started Guide · Read once, set up in ~15 minutes, file clean tickets.

TRUST

Safety & security, built in

Not add-ons: these are on by default, and they're why teams trust it with real tickets.

We never see your work

BugIt runs in your editor; your tickets go only to the AI model and tracker you connect. The only thing that ever reaches us is license/update data: your BugIt account sign-in (in the browser, on the Portal), a one-way hashed device ID, a random installation identifier, your device label (hostname) and operating-system name, the BugIt version and the plan filter you pick, short-lived cryptographic challenge material, the one-time approval token from the Portal link your browser opened, a per-activation acknowledgement secret (its hash, and the value itself only if you withdraw this device's access), and the Solo or Team entitlement you approve for this device. There is no license key. No telemetry, no analytics, ever.

Automatic backups

Before every update (and any time you ask) your config, glossary, house style, MCP endpoint settings, and learning are snapshotted on your device. Your signed device entitlement and credential values are excluded. One line brings them back: `Restore my settings`.

Nothing files without your yes

Every ticket and comment is previewed; an irreversible action happens only when you reply with exactly `FILE IT`: your whole message, tied to the draft you read, and usable once. BugIt can never delete a ticket, change its status, or change its assignee. Say `dry run` to practise: every external ticket, comment, edit, attachment, removal and notification is suppressed (explicit local commands such as backup or export can still create local files) and the agent refuses tracker writes, though that refusal follows BugIt's instructions rather than a platform/OS lock, so use read-only credentials when evaluating.

Signed, verified updates

Updates are cryptographically signed with a key kept *separate* from licensing. The agent verifies every byte and refuses anything tampered, then keeps all your settings.

No readable secrets on disk

Tokens go to your OS credential store and are never readable on disk: on Windows that is a record encrypted with your own Windows account's key, and on macOS and Linux the secret itself sits in the Keychain or the libsecret service. `config.json` holds only orgs and URLs, and a validator flags anything that looks like a leaked secret before it's ever saved, in config or in a command the agent runs for you.

Prompt-injection resistant

Text pulled from pages or tickets is treated as data, never as commands: injected instructions are flagged, not obeyed, and surfaced to you.

Customize safely

`config.json`, house-style, glossary, and MCP setup are yours, kept through every update. The agent file and other `.github/instructions/` files aren't: hand-editing those gets silently overwritten with no backup to restore from.

Private by design. Your bug reports, specs, glossary, and settings stay yours. Your work goes only to the tools you connected and your own AI model, never to us. The license and update data listed above is the only thing BugIt ever sends.

START HERE

60-second glossary

No tech background needed. Here are the only words that matter, in plain English.

VS Code

A free Microsoft app, the "window" this agent lives in.

GitHub Copilot

The AI brain that writes reports. You pay for a 1-year BugIt license (no subscription) plus either a Copilot plan or your own AI-provider usage.

BugIt

Your assistant in that window. Chat like texting a coworker.

Tracker

Where your team keeps bugs (Jira, ADO, GitHub...).

API token

A private password you create in your own tracker account, so BugIt can file as you. Saved in your operating system's credential store, encrypted with your own account's key on Windows and held in the Keychain or libsecret on macOS and Linux. Copying the folder to another machine copies nothing usable.

MCP server

An optional bridge for reading extra context (Confluence, Sentry, Notion). Filing does not need one.

config.json

A settings sheet. The agent fills it for you.

Chat

Where you type: "Write a bug report: app crashes on save."

If you can install an app, sign in, and type a sentence, you can run this.

COST & LICENSING

What it costs

Works with your favorite AI model. BugIt steers the model inside GitHub Copilot (GPT, Claude, or Gemini), or runs standalone with your own key.

Free

Copilot Free tier covers casual use.

No API keys

No hosting, no per-bug fees.

Copilot Pro

GitHub's paid tier for daily use.

No Copilot?

Bring your own OpenAI/Anthropic key.

$59.99

Solo, a one-year license with no subscription and no auto renewal, one device at a time, all features. **Team is available: $249.99, a one-year license with no subscription, for 5 member seats**, managed centrally in the BugIt Portal. Need more than five? Add a second Team license to the same Team and the seats add up.

That's a **1-year license with no subscription**, with no monthly billing and no auto-renew. Rival QA tools bill their price **every month**; save the time of a **handful of bugs** and it's paid for itself.

Team License

Share your tracker setup, glossary, and bug style with the whole Team: run `share.py export` once, and teammates preview the bundle with `import` which changes nothing. `--apply` adopts the settings; `--apply --apply-instructions` adopts the glossary and bug style too, so everyone files in the same format. In the BugIt Portal you invite people by email, give them the Admin or Member role, see who has activated a device, and nominate someone who can take over the Team if you are unavailable. Each member signs in with their own account and uses their own device: there is no shared key and no shared login.

WHY IT'S WORTH IT

What you actually get

Not "AI that writes bugs", but a QA agent that fits how your team already works.

WITHOUT BUGIT	WITH BUGIT
Write every bug by hand	One sentence → a complete, reviewed ticket
Duplicates pile up on the board	Auto dupe-check before you file, even if worded differently
Every tester's format differs	One consistent template, every time
Translate & look up specs by hand	Multilingual, and it pulls the context for you

Works with the tracker your team already uses

BugIt writes the ticket for any tracker and files it through the connector you enable. Switch anytime.

Files with a credential you create in your own account, checked against your destination before it is saved:

Jira Cloud · Azure DevOps · GitHub · GitLab · Bugzilla · YouTrack · Linear · Shortcut · ClickUp · Asana · Trello

Guided knowledge source (Atlassian Rovo MCP):

Confluence Cloud

Experimental, a known endpoint BugIt sets up and verifies live on your account:

Sentry Notion

Tells your team a bug was filed, built in: Slack takes a bot token and a channel, Microsoft Teams a Workflows URL, Discord a webhook URL, and email your SMTP settings. BugIt sends a real test message before saving the setting.

Slack Microsoft Teams Discord Email

Setup guidance only, via your own compatible MCP server:

Crashlytics BugSnag TestRail Xray

Not supported by automated setup:

S3 Google Drive Azure Blob

> **It also** checks duplicates, looks up crashes, pulls spec context, translates across languages, and writes verify/close comments.

SETUP

Up and running in 6 steps

One-time setup inside VS Code. Setup checks what you need and asks before it installs anything. BugIt needs no third-party Python packages.

1. **Install VS Code**
   Download from code.visualstudio.com and open it.
2. **Install an AI assistant**
   One that can run a command, such as GitHub Copilot Chat or the Claude extension: Extensions → Install → sign in.
3. **Open this folder**
   File → Open Folder. The agent loads from `.github/agents/`.
4. **Pick the agent**
   In chat, switch the mode dropdown to **BugIt QA Agent**.
5. **Type "Begin setup"**
   Answer a few questions; the agent writes `config.json`. If it's ever interrupted, saying it again resumes instead of starting over.
6. **Activate, then connect your tracker**
   Type Activate, sign in to the BugIt Portal in your browser, and approve this device; the command then saves the authorization and finishes. There is no key to paste. Then run `python tools/connect.py jira` once to save your own API token.

Shortcuts: `preset.py game` to start from a template · `setup_wizard.py` if no Copilot · `doctor.py` for a readiness checklist.

A complete step-by-step manual is included.

After you buy, you'll get a detailed, click-by-click setup & user guide (in English) written for people who've never done this before, so setup is easy and you're never stuck.

BEFORE YOU BUY: WHAT YOU'LL NEED

REQUIREMENT	WHY
VS Code (free)	The agent runs inside it
GitHub Copilot *(recommended)*	Powers the agent, or bring your own OpenAI/Anthropic key
Python 3.10-3.14	Runs the helper tools
A tracker account	Where bugs go. BugIt files to eleven trackers: Jira Cloud, Azure DevOps, GitHub Issues, GitLab Issues, Bugzilla, YouTrack, Linear, Shortcut, ClickUp, Asana and Trello. Each has guided setup, each writes with a credential you create in your own account, and BugIt checks that credential against your destination before saving it

Good to know

BugIt runs in **VS Code with GitHub Copilot**, in the **Claude extension**, and in a plain terminal; it steers **your** AI model rather than shipping one; it files to all **eleven** trackers it names, each with a credential you create in your own account, though what reaches a sortable priority field differs per tracker and three of them have no such field; Confluence, Sentry and Notion are read only, and they are the only things that need one; Slack, Teams, Discord and email notifications are built in; and redaction and other safeguards are aids, not guarantees: **you review and approve every ticket**, and results are more consistent on a stronger AI model than a very lightweight/free one.

SEE IT WORK

One sentence in. A clean ticket out.

This is the whole point. Type it like you'd say it to a coworker; the agent does the rest.

YOU TYPE

```
"the app crashes every time I tap Save on iPhone"
```

↓

BUGIT DRAFTS IT: YOU REVIEW, THEN TYPE "FILE IT"

[Crash] App closes when tapping Save (iPhone)

Steps	1. Open the app 2. Make any change 3. Tap **Save**
Expected	The change saves and the app stays open
Actual	The app closes instantly, every time
Details	Repro **5/5** · Severity **Critical** · Platform **iOS (iPhone)**

Then it dedupe-checks and files it to your tracker once you reply `FILE IT` (setting the tracker's own priority field to match where the tracker has one, and writing the severity into the ticket where it does not), and hands you the link. Want to tweak first? Just say *"make severity high"*. Want to practice? Say `dry run` and it writes the whole thing without filing.

EVERYDAY, YOU JUST TYPE

YOU TYPE	YOU GET
`Write a bug report: …`	Full report, dupe check, version auto-detect
`Close #ID` · `Translate to ja: …`	Verify/close comment · glossary-consistent translation
`Update` · `Back up my settings` · `help`	Signed update · local snapshot · full command list

OPT-IN EXTRAS

Power features explained

Turn these on in `Begin setup`, then trigger them from chat.

Triage digest

Type `Triage digest` for an instant standup summary: what was filed/changed, counts by severity and category.

Export bug to file

`Export bug to file` saves the report as local Markdown/JSON instead of filing. Ideal offline or to keep a copy.

Undo last filing

`Undo last filing` shows the last write in BugIt's audit log and explains what recovery is available. BugIt cannot delete or close a ticket and never invents a retraction note. After a fresh preview and `FILE IT` it can remove only a comment or attachment it created itself.

SLA flags

High-severity bugs get an SLA/due marker as they're filed, so urgent issues stand out automatically.

Per-category fields

Each category adds its own fields (crash → stack trace, UI → screenshot) so every report captures the right detail.

All optional. Skip them and the core write→review→file loop still works perfectly.

Licensing that respects you: after a successful online activation, a cached lease keeps this device working offline for up to 72 hours, after which it re-verifies online. One device at a time. When you move to a new device, the old one stops being able to renew straight away, and its slot is freed as soon as it reconnects or its current 72-hour lease runs out.

SUPPORT

Troubleshooting

SYMPTOM	FIX
"No tracker enabled"	Run `Begin setup`, pick a tracker
Filing is refused	Run `python tools/connect.py jira --check` and read the `fixes` it lists
Asks to re-activate	Type `Activate` and complete the browser sign-in. BugIt never displays saved credential values.
Won't file	Check `validate_config.py --test`; confirm primary tracker + signed in
Looks game-specific	Edit `config.categories` to your domain

Still stuck? Visit **bugit.dev** or email **support@bugit.dev** (support is handled in English), with real human help. Setup trouble in your first 7 days? We'll get you running or help with a store refund.

A whole year, for one payment

$59.99 is a 1-year license with no subscription and no auto-renew, and the year starts on the day you buy. There are no per-bug fees and no monthly billing. Rival QA tools bill their price **every month**. Save the time of just a handful of bugs and it's paid for itself.

~10 min saved/bug

one line → full ticket

Dupe check

before you file

Works with your tracker

the one you already use

Multilingual

10+ languages, glossary-locked

*I built **BugIt** because after years of writing and reviewing bug reports every day, I got tired of spending time on repetitive formatting instead of finding real issues. BugIt is the tool I wished I'd had.*

Languages: en, ja, fr, de, es, pt-br, it, ko, zh, ru, ar & more · Solo $59.99 · Team $249.99 · 1-year license, no subscription, 5 seats per Team license

BugIt · updates included while your license is active · Solo one user, Team up to five members (no resale or redistribution)
© 2026 Taskivator. All Rights Reserved. · bugit.dev