**Bug life cycle (Bug report)**

**A bug life cycle is the set of stages a software defect passes through from the moment it is reported until it is resolved and closed. The typical cycle starts when someone reports a new issue; the report is then triaged and assigned to a developer. The developer investigates and marks the issue Open while working on it. When a fix is implemented the status becomes Fixed. The QA/tester then retests the software: during Pending Retest the fix is tested. If the tester confirms the bug no longer occurs, the bug is marked Verified and then Closed. If the fix fails verification or the problem reappears, the bug is Reopened and the cycle repeats.**

**New (Reported): A tester, user, or developer files a bug with details: summary, steps to reproduce, expected versus actual behavior, screenshots/logs, severity, and priority.**

**Assigned: The project lead or triage team reviews the report and assigns it to the appropriate developer or team for investigation. Assignment confirms responsibility.**

**Open (In progress): The developer investigates root cause, may add comments, reproduce locally, and write or modify code to fix the defect. Any changes are recorded in the bug report.**

**Fixed: The developer commits the code and marks the bug as fixed. A reference to the change (commit ID or patch) is often included. The fix is ready for QA verification.**

**Pending Retest: The QA team runs the test cases and steps to reproduce the bug to confirm whether the fix works and that it didn’t cause regressions.**

**Verified: QA confirms the issue is resolved in the tested build/version. If verified, the report moves to closure.**

**Closed / Reopened: If QA verification is successful and no related issues remain, the bug is Closed. If the issue persists or reappears, the bug is Reopened and goes back to Assigned/Open for further work.**

**This lifecycle ensures traceability (who fixed what and when), reproducibility (how a bug was seen), and quality control (verification before closure). Good bug reports and disciplined lifecycle management reduce rework and speed up delivery.**

**References**

* **Bugzilla Foundation. (n.d.). *Bugzilla Documentation*. Retrieved from https://www.bugzilla.org/docs/**
* **Sommerville, I. (2016). *Software Engineering* (10th ed.). Pearson.**