Smoke testing (software)

In computer programming and software testing, **smoke testing** (also **confidence testing**, **sanity testing**,[1] **build verification test** (**BVT**)[2][3][4] and **build acceptance test**) is preliminary testing to reveal simple failures severe enough to, for example, reject a prospective software release. Smoke tests are a subset of test cases that cover the most important functionality of a component or system, used to aid assessment if main functions of the software appear to work correctly.[1][2] When used to determine if a computer program should be subjected to further, more fine-grained testing, a smoke test may be called an **intake test**.[1] Alternately, it is a set of tests run on each new build of a product to verify that the build is testable before the build is released into the hands of the test team.[5] In the DevOps paradigm, use of a BVT step is one hallmark of the continuous integration maturity stage.[6]

For example, a smoke test may address basic questions like "does the program run?", "does the user interface open?", or "does clicking the main button do anything?" The process of smoke testing aims to determine whether the application is so badly broken as to make further immediate testing unnecessary. As the book *Lessons Learned in Software Testing*[7] puts it, "smoke tests broadly cover product features in a limited time ... if key features don't work or if key bugs haven't yet been fixed, your team won't waste further time installing or testing".[3]

Smoke tests frequently run quickly, giving benefits of faster feedback, rather than running more extensive test suites, which would naturally take much longer.

A daily build and smoke test is among industry best practices.[8] Smoke testing is also done by testers before accepting a build for further testing. Microsoft claims that after code reviews, "*smoke testing* is the most cost-effective method for identifying and fixing defects in software".[9]

One can perform smoke tests either manually or using an automated tool. In the case of automated tools, the process that generates the build will often initiate the testing.

Smoke tests can be broadly categorized as functional tests or as unit tests. Functional tests exercise the complete program with various inputs. Unit tests exercise individual functions, subroutines, or object methods. Functional tests may comprise a scripted series of program inputs, possibly even with an automated mechanism for controlling mouse movements. Unit tests can be implemented either as separate functions within the code itself, or else as a driver layer that links to the code without altering the code being tested.

See also

- "Hello, World!" program
- Sanity testing
- Shakedown

References

1. ISTQB® Glossary for the International Software Testing Qualification Board® software testing qualification scheme, ISTQB Glossary (https://www.istqb.org/downloads/glossary.html) International Software Testing Qualification Board.
2. Dustin, Rashka, Paul. "Automated Software Testing -Introduction, Management, and Performance". Addison-Wesley 1999, p. 43-44. ISBN 0-201-43287-0.
3. Kaner, Bach, and Pettichord. "Lessons Learned in Software Testing". Wiley Computer Publishing, 2002, p. 95. ISBN 0-471-08112-4
4. "How to: Configure and Run Build Verification Tests (BVTs)" (https://msdn.microsoft.com/en-us/library/ms182465%28v=VS.80%29.aspx). *MSDN Library for Visual Studio 2005*. Retrieved 2010-11-20.
5. Samuel Menaker; Sheetal Guttigoli (14 December 2014). *Managing Software Development* (https://books.google.com/books?id=kuXXBQAAQBAJ&pg=PA40). Samuel Menaker, Sheetal Guttigoli. p. 40. GGKEY:JH61NP21TXJ.
6. PowerShell Magazine (http://www.powershellmagazine.com/2016/01/05/devops-infrastructure-as-code-and-powershell-dsc-the-introduction/), DevOps, Infrastructure as Code, and PowerShell DSC: The Introduction, by Ravikanth C, January 5, 2016

7. Cem Kaner, James Bach, Bret Pettichord, *Lessons learned in software testing: a context-driven approach*. Wiley, 2001
8. McConnell, Steve. "Rapid Development". Microsoft Press, p. 405
9. "Guidelines for Smoke Testing" (https://msdn.microsoft.com/en-us/library/ms182613(VS.80).aspx). *MSDN Library for Visual Studio 2005*. Retrieved 2010-11-20.

External links

- PC Mag's Definition (https://www.pcmag.com/encyclopedia_term/0,2542,t=smoke+test&i=51556,00.asp)

Retrieved from "https://en.wikipedia.org/w/index.php?title=Smoke_testing_(software)&oldid=869215972"