

Top 10 reasons why there are Bugs in Software!

Many of the testers are thinking that *why these bugs are introduced* in the code or why developer leaves the bugs in the code. I think this is contradictory type of question means if no **introduction of errors** in the code then there is no need of testing of software, just develop the defect free software application & use it without testing. This is not practicable, if the complexity of the software application increased then it consider that the defects will introduced in the code. There are few factors which are the preliminary causes of introduction of the defects in code. So in this article I will discuss about what all probable reasons which may cause the defects in the software? Also I am talking about top 10 possible causes of errors, defects and bugs in software. Let's see what all reasons of introduction of Defects/Bugs in the code:

- Miscommunication of requirements introduces error in code
- Unrealistic time schedule for development
- Lack of designing experience
- Lack of coding practices experience
- Human factors introduces errors in code
- Lack of version control
- Buggy third-party tools
- Last minute changes in the requirement introduce error
- Poor Software testing skill

So here I will take all above above points one by one & explain in detailed:



1) Miscommunication of requirements introduces error in code: Miscommunication of the requirements is one the most common problem in the software development process which causes an introduction of defects in the code. It means erroneous & lack of communication in the software development process. This may introduce from the first step of SDLC means from requirement gathering to development of the product. If the requirements are incomplete or vague then the development leads to situation where developers facing in developing software application based on incomplete requirements and this cause the issue in testing of such incomplete application. Also communication errors are introduced in case if application developed by other developer & modified by other developer.

2) Unrealistic time schedule for development: In the fast moving world Stakeholders want new things implemented before the competitors implement it. So to develop and QAed the application in unrealistic project deadlines make impact in the quality of the project & cause to introduce the defects in the application. It should not be developed under crazy release schedules with the insufficient or limited resources create bad impact. Many of the times developers not get enough time to design, develop & unit tested the application in dev environment prior to move to tester's plate to test complete application. So more likely the last minutes change made could introduce errors in the code.

3) Lack of designing experience: To start with development the main core part of development is design & how good your design is decides the overall software application development. In the current age of complex software development market either implementation is complex or to implement the project more research & development required. Sometimes it also complicated to understand the problems. Sometimes they have to complete the project in a specific time period, so doing design, development & testing in specific period time set may cause the errors.

4) Lack of coding practices experience: Coding plays an important role in Software development process. Simply bad coding leads to errors into code. Bad coding means unhandled exceptions, errors, improper validations of inputs. Few programmers are working with bad tools like faulty debuggers, compilers, validators etc., using this is about to add errors in the code due to tricky to debug the code.

5) Human factors introduces errors in code: Most common area to get defects introduced in the code is the "Human Errors" this is because of the system is developed by the human being & they are not perfect all the time. As system is developed by the human so we cannot expect the system would be bug free, hence this is the one area from where the errors might be introduced in the system. If paradoxically the we are yet to find out the non-human who can develop the software for us. So we have to rely on the human being to develop the software.

6) Lack of version control: If the functionality was tested in previous builds & after next few build it is seeing that regression errors occurs so it is very difficult to identify the regression errors. So in such a case version controlling makes a key role. Even if a version control system (e.g. Visual SourceSafe) is in place, errors might still slip into the final builds if the programmers fail to make sure that the most recent version of each module are linked when a new version is being built to be tested.

7) Buggy third-party tools: Most of the time we have to use [Third party tools](#) in software development process. These are open source tools or commercial tools based on project requirement. Tools like Compiler, Debugger, HTML Editors, plug-ins, external APIs, class libraries, Shared DLLs are developed by someone else & might have bugs in the tools which adding error in our software.

8) Last minute changes in the requirement introduce error: Requirements changes in the last minute can be dangerous which results instability of software application. To implement the changes which are introduced in 11th hour requirement changes will definitely introduce errors in the application & also existing working feature may stop working.

9) Poor Software testing skill: In many organizations poor testing can be executed, this would not like to see by any tester but let's face it. There can be insufficient knowledge of testing skills (Lack of skilled testing) which leads to defects in the system. Also if tester is not paying attention in testing activity & doing testing without giving importance to it, so it results poor quality of the product & major bugs remains in the software. Moreover, in this era of agile software development poor unit tests (e.g. in TDD) may result in poor coding and hence escalate the risk of errors.

10) Next to you: This last point is intentionally left blank, so why don't you come up with one **reason why there are Bugs/Defects** in Software. Feel free to add queries or any things related to testing in comments below, I will definitely address questions as early as possible. If you like the article so take a step to share the article over your friends using below popular sharing options.

Source:

<http://www.softwaretestingclass.com/top-10-reasons-why-there-are-bugs-defects-in-software/>