

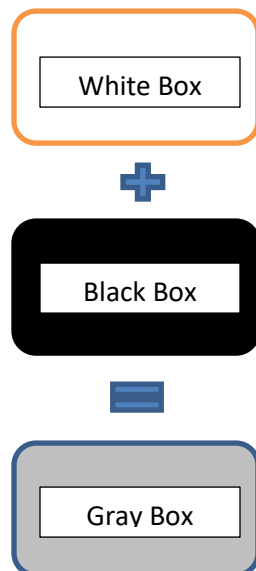
GRAY BOX TESTING

What is Gray Box Testing

To understand what gray box testing means, first we need to understand what software testing means! Software Testing is an activity to check whether the output / result is equivalent to the expected output / result, which means that the software is correctly running. The result which is obtained after certain software / system is run, must match with the result which is expected as the output from the software / system; if it fails to do so, the software needs to be re-written or certain changes need to be done inside it. To define it in simple terms, Gray Box Testing is an amalgamation of Black Box Testing and White Box Testing. Gray Box Testers input data from User Interface and check internally whether that data is flowing through the logic or the code in a defined manner.

Understanding Gray Box Testing

Gray box Testing is a product of Black Box Testing and White box Testing. Black Box Testing means the tester has no knowledge of how the software works inside. This type of testing is done at the user level. So the tester checks whether the end result is obtained, and doesn't know whether the code is working correctly at the loops and the breaks inside. So, the Software Testers are the ones who generally are responsible for Black Box Testing. Exactly opposite is White Box Testing. This type of testing is done mostly by the Software Developers as they check whether a particular result is being obtained at a particular break. In white box testing, the testers (generally developers) have the knowledge, of how the software is internally working.



As told earlier, in gray box testing, for testing purposes, tester enters some dummy values to check the correct flow of the output. So for using Gray Box Testing the tester needs to have knowledge of both software development and testing, so as to check the correct flow.

Examples of Gray Box Testing

As we know by now, that only a partial part of the logic is known to the tester in gray box testing; it becomes a middle way in which the user can test the logic or software. The best example to explain the same would be; in a certain software, the user needs to use some third party application. This application, when is being used, only a part of it is exposed to the developer. So, now this can only be checked using the input data and some of the part which has been exposed. This is an perfect example of how Gray Box Testing works.

Another example would be the working of HTML links. The tester checks for the links. Some of the links, he may click, may or may not open the correct page. When the link does not go to the expected page, the tester can change the link address from the partially exposed code and correct it.

One more example of Gray Box Testing is the validations which are used while inputting data. Most of us, have experienced this, while entering details online, we get errors if we input some wrong data, example; "abc@con.con". Now, we will see this wrong input and get the error message. Tester will rectify this at their end by disabling the code.

Gray Box Testing Techniques

1. **Matrix Testing:** Developers define the entire variable that might be used during the execution of the software. Each of these variables has a technical and business risk associated with it. The risks are tested under matrix testing phase.
2. **Pattern Testing:** An analysis is done of the previous failures of the softwares. Why and how the software has failed is taken into account and logged for future references. This helps in designing test cases in future, which will not let the software fail.
3. **Orthogonal Testing:** Usually used, when the amount of data is less but the complexity is more. So, all the possible permutations and combinations are used to assess.
4. **Regression Testing:** When certain changes are done in the software to get the desired output, regression testing is done, to check whether the current logic is not affecting the output and working with the software and desired result is still being derived.

Advantages of Gray Box Testing

1. As is a derivative of Black Box and White Box Testing methods, it adds more of both the the testing techniques advantages.
2. Testing is done from more of user perspective than of developers perspective.
3. The testers need not have the access to all of the code / logic.
4. Instant fixes can be done, as a partial code is available.
5. Flow of the data is managed and maintained correctly.
6. Fair review of the software is done, and not conflicts arise between developers and testers.

Disadvantages of Gray Box Testing

1. As only limited access of code / logic is available, complete fixes cannot be done sometimes, which means, sometimes the software can remain as it is.
2. Other white box testing types like algorithm testing cannot be done, as complete logic is not available.
3. Difficult to perform this type of testing on distributed architected software systems.

Why should we use Gray Box Testing

As of now, we all know that it is very effective with not only web applications but with business applications too, so it'll rectify the most of the software solutions. As the name goes, Gray Box sometimes also known as transparent box, the tester need not have the full understanding of the system. This testing method is certainly penetrates through the application and gets to the core of the problem, and without knowledge of the whole code, it can be fixed.

Conclusion

With so many usage advantages, one will necessarily require Gray Box Testing. And as specified earlier a combination of both testing methods, Gray Box Testing is certainly an effective technique to find out the errors in the software system.